



FACTULTAD DE PSICOLOGÍA

DEPARTAMENTO DE PSICOLOGÍA BÁSICA

TESIS DOCTORAL

PENSAMIENTO COMPUTACIONAL Y RESOLUCIÓN DE PROBLEMAS

Autora:

BEATRIZ ORTEGA RUIPÉREZ

Director:

MIKEL ASENSIO BROUARD

Madrid, 2017

El objetivo de esta tesis es contribuir al conocimiento del pensamiento computacional, profundizando en cómo este pensamiento mejora la resolución de problemas complejos. Para ello, se ha definido de forma operativa este pensamiento en términos de procesos cognitivos, basándonos en la literatura revisada sobre el tema. De esta forma, se ha creado un marco de evaluación de este pensamiento, de cara a poder comprobar las dos principales partes de la resolución de problemas en las que este pensamiento puede ser de mayor beneficio, es decir, cómo el pensamiento computacional facilita la representación, o adquisición de conocimiento sobre el problema, y cómo facilita el proceso de resolución, o aplicación del conocimiento, en la resolución de problemas complejos.

Esta investigación emplea una metodología cuantitativa, empleando un estudio pre-post comparando los resultados entre un grupo que participa en una formación sobre programación y pensamiento computacional, y un grupo control que no participa en ninguna formación alternativa. De esta forma, podremos conocer el alcance del llamado pensamiento computacional, estudiando la relación entre el uso de las estrategias utilizadas por los científicos de la computación y la resolución de problemas complejos. Para medir la resolución de problemas complejos, se ha creado y validado un instrumento de evaluación consistente en unas pruebas interactivas basadas en el enfoque de sistemas complejos múltiples, adoptado por las pruebas PISA 2012. Estas pruebas se basan en dicho enfoque y establecen un marco para medir 4 fases de la resolución de problemas complejos en diferentes niveles. Entre las pruebas del instrumento se encuentran dos de utilizadas en PISA 2012, para comparar los resultados y asegurar dicho enfoque.

Los resultados obtenidos en el estudio realizado para esta tesis, nos han permitido profundizar en el conocimiento sobre el pensamiento computacional, en cómo está relacionado con la resolución de problemas complejos. Por una parte, confirmamos que la relación entre pensamiento computacional y resolución de problemas complejos es cierta. Por otra parte, comprobamos que un desarrollo inicial de este pensamiento facilita el proceso de resolución de problemas, es decir, ayuda a la planificación y ejecución de este proceso, y a la observación y reflexión sobre si este proceso es adecuado; y en menor medida también facilita la representación del problema, aunque en este caso se sugiere que es necesario un mayor desarrollo de este pensamiento para facilitar la exploración y entendimiento del problema que nos lleva a su representación y formulación.

Como conclusión obtenemos que el pensamiento computacional facilita el proceso de resolución de problemas, y en niveles superiores, facilita la representación del problema.

A mi familia

Agradecimientos

Esta tesis ha podido realizarse gracias al apoyo de todas aquellas personas que han estado conmigo, de una forma u otra, a lo largo de todos esos momentos tan diferentes que uno se encuentra a lo largo de la realización de este largo trabajo. Sin ellas, no habría llegado hasta aquí, y es por ello que debo agradecerse a cada una de ellas.

En primer lugar, dar las gracias a mi director, Mikel Asensio, por haber estado ahí siempre que lo he necesitado, con mis continuos cambios y redefiniciones de la tesis durante estos casi tres años. Por ayudarme a ver las cosas de forma sencilla y abordable, cuando para mí eran un gran obstáculo que me impedía ver hacia dónde iba y me daban ganas de abandonar. Por darme esa seguridad en mí misma y mi trabajo, que sin ella esto no hubiese sido posible.

A mis excompañeros de BQ, con quienes siempre podía comentar todo lo que me iba pasando en ese largo camino, y quienes me escuchaban con entusiasmo a pesar de no enterarse de nada sobre lo que les estaba hablando. En especial a Alfredo Sánchez, quien desde el principio ha estado ahí y que ha estado conmigo hasta el final de esta tesis, ayudándome en lo que ha podido, que ha sido mucho.

Al Colegio Hélicon, en especial a Álvaro, Alicia, Laura y Loli, que me han ayudado siempre que se lo he pedido y gracias a ellos he podido validar las pruebas de la investigación, a pesar de volverles locos al desistir de mi primer intento de investigación.

Agradecer a todas las personas que han estado ahí estos tres años, con los que he podido pasar buenos ratos y hacer que estos años no hayan sido tan duros. A las chicas del norte, a Sara, Tere, Verge, Bea, Mery y Marta. A los chicos del crespó, a Sergio, Deinar, Raúl, Diego, Chechu.

A mis amigos de la facultad, a los que le debo parte de cómo soy. Por esos cinco años tan importantes de mi vida que me han hecho madurar hacia lo que soy ahora. A Lucía, María, Alba, Rosana, Rubén, Marcos, Edu, Maribel, Sara y Carol; con los que he compartido tantos momentos y a los que echo de menos ahora que estamos tan ocupados con nuestras vidas.

Y lo más importante, agradecer a mi familia todo su apoyo. A los que ya no están, mis abuelos, que me han enseñado tanto y de los que me sigo acordando continuamente. A todos mis tíos y todos mis primos, gracias por ser como sois, tan alegres y con tantas energías que se contagian enseguida, gracias por todos los momentos que hemos pasado juntos, que seguimos pasando y que seguro que seguiremos pasando juntos.

Agradecer especialmente a mis padres, Amparo y Mariano, que tanto me han dado y me siguen dando, gracias por aguantarme todos estos años, y por haberme hecho ser lo que soy. Gracias también a mis hermanas, Carol y Belén, que desde pequeña han sido mi modelo a seguir, por haberme enseñado tanto y haberme hecho pensar más allá. Gracias a mis cuñados, Ángel y Álvaro, quienes son parte de la familia. Y gracias a mis sobrinos, a Lucía, con la que estoy volviendo a disfrutar como una enana, y a los pequeños Pablo y Adriana, que todavía son muy pequeños pero que parece que llevan aquí toda la vida.

Y, por último, pero no menos importante, a Alejandro, mi pareja, con la que paso todo el tiempo que puedo, que ha estado ahí siempre que lo he necesitado. Gracias por aguantarme durante todo este tiempo, por animarme cuando parecía que todo estaba perdido y por ayudarme a encontrar solución a mis problemas. Gracias por ayudarme en todo lo que has podido, incluido el tener que crearme las pruebas de la tesis durante dos meses en tus ratos libres. Gracias por esto y mucho más.

Gracias a todos los que he mencionado, y también a los que se me haya podido olvidar mencionar, por estar ahí de una u otra forma, y haberme ayudado a conseguirlo.

Índice de contenidos

Prólogo	25
BLOQUE 1. MARCO TEÓRICO	35
Capítulo 1. Qué es el pensamiento computacional	37
Qué se entiende por pensamiento computacional	37
Intentos de una definición consensuada	43
PC como herramienta cognitiva	45
Relación entre PC y las ciencias de la computación	46
Definición operativa del PC	51
Relación entre pensamiento computacional y resolución de problemas	59
Capítulo 2. Resolución de problemas complejos	65
Estado actual de la investigación sobre la resolución de problemas	65
Representación del problema	67
Proceso para solucionar el problema	83
Resolución de problemas complejos (RPC)	88
Características de la RPC	90
Proceso de la RPC	92
Capítulo 3. Aportaciones del pensamiento computacional a la RPC	95
Aproximaciones del pensamiento computacional en la resolución de...	95
Definición operativa del Pc y la resolución de problemas complejos	96
Procesos compartidos con la resolución de problemas	97
Abstracción	97
Tratamiento de datos: recopilación, análisis y representación	98
Algoritmos o secuencialización	101
Procesos específicos del PC	102
Descomposición del problema	103
Automatización	104
Paralelismo	105
Simulación	106
PC como facilitador de la RPC	107

Capítulo 4. Investigación sobre resolución de problemas complejos y pensamiento computacional	109
Evolución en la investigación sobre resolución de problemas	109
Corrientes en la investigación de resolución de problemas	110
Técnicas tradicionales de investigación en la resolución de...	110
Investigación de la resolución de problemas complejos	111
Uso de marcos formales para el estudio de la RPC	113
Características de la RPC y los marcos formales	114
Problemas utilizados en la RPC	116
Investigación sobre pensamiento computacional	123
BLOQUE 2. ESTUDIO EMPÍRICO	129
Capítulo 5. Método de la investigación	131
Diseño experimental e hipótesis de estudio	131
Participantes	134
Instrumentos y materiales	135
Pruebas de evaluación para la resolución de problemas	135
Curso de pensamiento computacional	155
Procedimiento	157
Capítulo 6. Validación de las pruebas de RPC	161
Evaluación del PC en la resolución de problemas complejos	164
Método para la validación	166
Resultados	193
Discusión	196
Capítulo 7. Resultados y discusión	199
Contraste de Bondad de ajuste	200
Homocedasticidad de la muestra	202
Contraste de hipótesis estadísticas	203
H1. El PC facilita la RPC	204
H2. El PC facilita la adquisición de conocimiento sobre el...	215
H3. El PC facilita el proceso de resolución del problema	220
Discusión de los resultados	225

BLOQUE 3. CONCLUSIONES GENERALES	227
Capítulo 8. Conclusiones y futuras líneas de investigación	229
Interpretación de las hipótesis de estudio	233
Limitaciones de la investigación	238
Conclusiones generales: discusión de las preguntas fundamentales...	239
Futuras líneas de investigación	242
Investigaciones relacionadas con la representación	242
Investigaciones relacionadas con las estrategias de resolución...	243
Otras posibles investigaciones	244
Impacto de esta investigación	245
Impacto en el campo del pensamiento computacional	245
Impacto en el campo de resolución de problemas complejos	247
Referencias	251
Anexos	269

Índice de tablas

	Pág.
Capítulo 4	
Tabla 1. Las 5 fases de MicroDYN y su relación con las 5 características de la RPC. Fuente Greiff y Fischer, 2013, p.50	117
Capítulo 6	
Tabla 2. Modelo PISA 2012 para la resolución de problemas complejos (OECD,2010), basado en el enfoque de sistemas múltiples complejos de Greiff, Wüstenberg y Funke (2012).	170
Tabla 3. Relación entre las pruebas y las diferentes fases de resolución de problemas en los niveles evaluados. En naranja aparecen las preguntas de las pruebas utilizadas en PISA 2012. Fuente: elaboración propia.	171
Tabla 4. Contexto de las pruebas. Fuente: elaboración propia	172
Tabla 5. Alfa de Cronbach general de las pruebas.	193
Tabla 6. Alfa de Cronbach si se eliminase cada elemento.	194
Tabla 7. Alfa de Cronbach al eliminar la pregunta Coche 4.	195
Tabla 8. Alfa de Cronbach cuando se elimina la pregunta Climatizador 2.	195
Tabla 9. Alfa de Cronbach al eliminar la pregunta Agenda 1.	195
Tabla 10. Instrumento de evaluación de RCP sólo incluyendo las pruebas con las que alcanza un alfa de Cronbach de 0,7).	196
Capítulo 7	
Tabla 11. Prueba de Shapiro-Wilk para comprobar la bondad de ajuste.	200
Tabla 12. Estadísticas descriptivas de los grupos en las variables RP_pre y RP_pos de la primera hipótesis de la investigación	205
Tabla 13. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias	205
Tabla 14. Estadísticas descriptivas de los grupos en las variables RP_inf_pre, RP_inf_pos, RP_sup_pre y RP_sup_pos de la primera hipótesis de la investigación	208

Tabla 15. Prueba de homogeneidad de varianzas para las variables RP_inf_pre, RP_inf_pos; RP_sup_pre y RP_sup_pos del enfoque por dificultad de la primera hipótesis	208
Tabla 16. Prueba de ANOVA de un factor para comprobar si existen diferencias entre los grupos en las cuatro variables del enfoque por dificultad de la primera hipótesis	209
Tabla 17. Correlación de la aproximación general de RPC para el grupo experimental	210
Tabla 18. Correlación de la aproximación por dificultad de RPC para el grupo experimental	211
Tabla 19. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC gracias a la intervención (GE)	211
Tabla 20. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC para PCI y PCS gracias a la intervención (GE)	212
Tabla 21. Correlación de la aproximación general de RPC para el grupo control	213
Tabla 22. Correlación de la aproximación por dificultad de RPC para el grupo control	213
Tabla 23. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC que no se deba a la intervención (GC)	213
Tabla 24. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC para PCI y PCS que no se deba a la intervención (GC)	214
Tabla 25. Estadísticas descriptivas de los grupos en las variables Adq_pre y Adq_pos de la segunda hipótesis de la investigación	216
Tabla 26. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias para la segunda hipótesis	216
Tabla 27. Estadísticas descriptivas de los grupos en las variables Repres_pre y Repres_pos de la segunda hipótesis de la investigación	218
Tabla 28. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias para la hipótesis 2.1	219
Tabla 29. Estadísticas descriptivas de los grupos en las variables Aplic_pre y Aplic_pos de la tercera hipótesis de la investigación	221

Tabla 30. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias para la tercera hipótesis	221
Tabla 31. Estadísticas descriptivas de los grupos en las variables Plan_pre y Plan_pos de la segunda hipótesis de la investigación	223
Tabla 32. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias	224
Tabla 33. Resumen de los P-Valor obtenidos para las diferentes hipótesis en relación con las fases de RPC, tanto en el pretest como en el postest. Fuente: elaboración propia	225

Capítulo 8

Tabla 34. Resumen de los P-Valor obtenidos para las diferentes hipótesis en relación con las fases de RPC, tanto en el pretest como en el postest	235
Tabla 35. Prueba T de Student para la variable resolución de problemas general, antes de eliminar las preguntas que aumentaban nuestra consistencia interna.	238
Tabla 36. Prueba T de Student para la variable planificación en la resolución de problemas complejos, antes de eliminar las preguntas que aumentaban nuestra consistencia interna.	239

Índice de figuras

	Pág.
Capítulo 1	
Figura 1. Representación del proceso de resolución de problemas con los términos empleados en PISA 2012 (OECD, 2010). Fuente: elaboración propia	61
Figura 2. Representación del proceso de resolución de problemas con los términos empleados en el pensamiento computacional (CSTA y ISTE, 2011). Fuente: elaboración propia	62
Capítulo 2	
Figura 3. Aspectos estáticos de la interacción producida en una situación de resolución de un problema complejo. Fuente: Funke y Frensch (1995)	90
Capítulo 4	
Figura 4. Impresión de pantalla de la tarea “Entrenamiento de balonmano” de MicroDYN. Fuente: Wüstenberg et al., (2012)	118
Figura 5. Impresión de pantalla de una tarea de Genetics Lab. Fuente: Greiff, Fischer et al. (2013)	119
Figura 6. Impresión de pantalla de la tarea de “televisión” de MicroFIN. Fuente: Greiff, Fischer et al. (2013)	120
Figura 7. Fortalezas y limitaciones de los enfoques de evaluación, correspondientes a actividades de diseño: análisis de proyectos, entrevistas basadas en artefactos y diseño de escenarios. (cf. Brennan y Resnick, 2012)	125
Figura 8. Resultados sobre la carga cognitiva en el grupo entrenado en pensamiento computacional (CT-group en azul) y en el grupo no entrenado en pensamiento computacional (NCT-group en gris). Fuente Park, Song y Kim (2015)	127
Capítulo 5	
Figura 9. Enunciado y pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia	137
Figura 10. Interacción con la agenda y vista de las pegatinas de la pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia	138

Figura 11. Enunciado de la pregunta 2 de la prueba Agenda basada en el marco MicroFIN. Fuente: elaboración propia.	139
Figura 12. Enunciado y pregunta 1 de la prueba Vídeo con enfoque MicroFIN. Fuente: elaboración propia.	139
Figura 13. Interacción con el calendario de la pregunta 1 de la prueba Vídeo, basada en el enfoque MicroFIN. Fuente: elaboración propia	140
Figura 14. Enunciado de la pregunta 2 de la prueba Vídeo basada en el enfoque MicroFIN. Fuente: elaboración propia	141
Figura 15. Enunciado de la prueba Billetes de PISA. Fuente INEE (2013)	142
Figura 16. Enunciado de la pregunta 1 de la prueba Billetes de PISA. Fuente: INEE (2013)	143
Figura 17. Enunciado de la pregunta 2 de la prueba Billetes de PISA. Fuente: INEE (2013)	144
Figura 18. Enunciado de la pregunta 3 de la prueba Billetes de PISA. Fuente: INEE (2013)	144
Figura 19. Enunciado de la prueba Tren y de la pregunta 1, prueba basada en el marco MicroFIN. Fuente: elaboración propia.	145
Figura 20. Enunciado de la pregunta 2 de la prueba Tren basada en el marco MicroFIN. Fuente: elaboración propia.	146
Figura 21. Enunciado de la prueba climatizador de PISA. Fuente: INEE (2013)	147
Figura 22. Enunciado de la pregunta 1 de la prueba climatizador de PISA. Fuente: INEE (2013)	148
Figura 23. Enunciado de la pregunta 2 de la prueba climatizador de PISA. Fuente: INEE (2013)	148
Figura 24. Enunciado de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.	150
Figura 25. Enunciado de la pregunta 1 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.	151
Figura 26. Enunciado de la pregunta 2 de la prueba Coche basada en el marco MicroDYN. En negro aparecen las relaciones correctas y en rojo las otras posibles relaciones. Fuente: elaboración propia.	152
Figura 27. Enunciado de la pregunta 3 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.	153

Figura 28. Enunciado de la pregunta 4 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia. 154

Capítulo 6

Figura 29. Enunciado y pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia 173

Figura 30. Interacción con la agenda y vista de las pegatinas de la pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia 174

Figura 31. Enunciado de la pregunta 2 de la prueba Agenda basada en el marco MicroFIN. Fuente: elaboración propia. 175

Figura 32. Enunciado y pregunta 1 de la prueba Vídeo con enfoque MicroFIN. Fuente: elaboración propia. 176

Figura 33. Interacción con el calendario de la pregunta 1 de la prueba Vídeo, basada en el enfoque MicroFIN. Fuente: elaboración propia 177

Figura 34. Enunciado de la pregunta 2 de la prueba Vídeo basada en el enfoque MicroFIN. Fuente: elaboración propia 178

Figura 35. Enunciado de la prueba Billetes de PISA. Fuente INEE (2013) 179

Figura 36. Enunciado de la pregunta 1 de la prueba Billetes de PISA. Fuente: INEE (2013) 180

Figura 37. Enunciado de la pregunta 2 de la prueba Billetes de PISA. Fuente: INEE (2013) 180

Figura 38. Enunciado de la pregunta 3 de la prueba Billetes de PISA. Fuente: INEE (2013) 181

Figura 39. Enunciado de la prueba Tren y de la pregunta 1, prueba basada en el marco MicroFIN. Fuente: elaboración propia. 182

Figura 40. Enunciado de la pregunta 2 de la prueba Tren basada en el marco MicroFIN. Fuente: elaboración propia. 183

Figura 41. Enunciado de la prueba climatizador de PISA. Fuente: INEE (2013) 184

Figura 42. Enunciado de la pregunta 1 de la prueba climatizador de PISA. Fuente: INEE (2013) 184

Figura 43. Enunciado de la pregunta 2 de la prueba climatizador de PISA. Fuente: INEE (2013) 185

Figura 44. Enunciado de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.	187
Figura 45. Enunciado de la pregunta 1 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.	188
Figura 46. Enunciado de la pregunta 2 de la prueba Coche basada en el marco MicroDYN. En negro aparecen las relaciones correctas y en rojo las otras posibles relaciones. Fuente: elaboración propia.	189
Figura 47. Enunciado de la pregunta 3 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.	190
Figura 48. Enunciado de la pregunta 4 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.	191

Capítulo 7

Figura 49. Histograma de la muestra en el primer momento de evaluación: pretest	201
Figura 50. Histograma de la muestra en el segundo momento de evaluación: postest	201

Capítulo 8

Figura 51. Modelo de análisis para el estudio del pensamiento computacional y de la estructura. Fuente: elaboración propia	231
--	-----

Acrónimos

A continuación, se recogen los acrónimos utilizados a lo largo de todo el trabajo.

CC. Ciencias de la Computación

CSTA. Computer Science Teacher Association (Estados Unidos)

EEG. Electroencefalograma

ESO. Educación Secundaria Obligatoria

Et al. Y colaboradores

FSA. Automatas de estados finitos (finite state automata)

GC. Grupo control

GE. Grupo experimental

INEE. Instituto Nacional de Evaluación Educativa

ISTE. International Society Technology Education (Estados Unidos)

K-12. Kinder-12¹

LSE. Sistemas lineales de ecuaciones estructurales (linear structural equations)

MCS. Sistemas complejos multiples (multiple complex systems)

NRC. National Research Council (Estados Unidos)

OCDE. Organización para la Cooperación y el Desarrollo Económicos

¹ La K designa la educación desde el kindergarden (correspondiente a educación infantil). El 12 designa los años de educación obligatoria desde el kindergarden, correspondiente hasta el fin de la educación secundaria.

PC. Pensamiento computacional

PCI. Problemas complejos inferiores

PCS. Problemas complejos superiores

PISA. Programme for International Student Assessment

Q2L. Quest to Learn

RP. Resolución de problemas

RPC. Resolución de problemas complejos

SPSS. Statistical Package for the Social Sciences

STEM. Acrónimo (in): Science, Technology, Engineering, Mathematics.

TIC. Tecnologías de la información y la comunicación

TSP. Problema del viajante del comercio (Traveling Salesman Problem)

Ud. Unidad

Uds. Unidades

USA. Estados Unidos de América (United States of America)

Índice de anexos

Anexo 1. Distribución de los aciertos en cada pregunta (muestra validación).	269
Anexo 2. Distribución de los aciertos en cada pregunta (muestra pretest).	273
Anexo 3. Distribución de los aciertos en cada pregunta (ambas muestras).	277

Prólogo

Objetivo y descripción del estado actual del tema

El objetivo de esta tesis doctoral consiste en comprobar cuál es el alcance del empleo de un pensamiento computacional en la resolución de problemas complejos.

El pensamiento computacional (Wing, 2006) surge desde las Ciencias de la Computación, propuesto como una forma de pensamiento que permite abordar los problemas de forma computacional, y que influye en la planificación y ejecución de los problemas complejos; al utilizar técnicas que emplean habitualmente los científicos de la computación, facilitando la resolución de problemas.

La resolución de problemas implica una representación del problema y la construcción de un curso de acciones que pretenden lograr el objetivo representado (Holyoak y Morrison, 2012, p.2). Este curso o proceso de solución de problemas sigue una secuencia determinada desde el estado inicial del problema hasta alcanzar su solución. A partir de esta definición, y de la definición propuesta para el pensamiento computacional, la literatura revisada sugiere que el pensamiento computacional se aplica en el proceso de resolución de problemas.

Por otra parte, se han encontrado indicios (Kwisthout, 2012; Wing, 2014) de que podría ser necesario un modelo computacional para la representación de problema complejos, ya que sería más fácil gestionar gran cantidad de variables consideradas de una relevancia similar entre ellas, y se facilitaría así la resolución de problemas.

Al utilizar una estrategia computacional para construir esa representación, podríamos decir que el pensamiento computacional también facilita la representación de problemas complejos, y no sólo se aplica al proceso de resolución como se había planteado inicialmente.

Conexión biográfica e interés objetivo

Por qué es importante el pensamiento computacional

Wing subraya la importancia del pensamiento computacional (PC) cuando acuña este término por primera vez en 2006. A lo largo de sus publicaciones (Wing, 2006, 2008, 2011, 2014) ha continuado dando motivos por los que considerar el PC como un pensamiento fundamental. De estas publicaciones, queremos destacar lo siguiente:

El pensamiento computacional es descrito como un conjunto de habilidades y procesos cognitivos. Estos procesos y habilidades son indispensables para los expertos en Ciencias de la Computación (CC), y para otras disciplinas científicas. Por lo que también se considera que el pensamiento computacional es necesario para todas las personas, y no sólo para los científicos de la computación (Wing, 2006).

Si el pensamiento computacional consiste en una herramienta que se utiliza para resolver problemas complejos (Wing, 2006), la persona que emplea el pensamiento computacional puede descomponer el problema en pequeños problemas que sean más fáciles de resolver, y reformular cada uno de estos problemas para facilitar su solución por medio de estrategias de resolución de problemas familiares.

Por tanto, según la autora, este pensamiento es importante porque, usando conceptos computacionales, es decir, haciendo cálculos, podemos mejorar la forma en la que nos aproximamos a los problemas, gestionamos nuestra vida diaria y nos comunicamos e interactuamos con otras personas (Wing, 2006).

Wing (2014) señala que los beneficios educativos resultarían en ser capaz de pensar computacionalmente, empezando por el uso de abstracciones, mejorando las habilidades intelectuales. Esta capacidad de pensar de forma computacional puede ser transferida a cualquier dominio de conocimiento.

Este pensamiento, como ya hemos visto, podría ser un factor fundamental para saber emplear eficazmente los procesos científicos, es decir, saber resolver un problema utilizando un método riguroso y científico (National Research Council (NRC), 2010). El PC podría ser empleado en la investigación en ciencias y en humanidades (Wing, 2008).

Por otra parte, Wing (2008) resalta la importancia de este pensamiento en cuanto a las posibilidades que nos ofrece utilizar el pensamiento computacional en la computación. Gracias al PC podríamos diseñar y desarrollar sistemas cada vez más complejos. Esta evolución en la complejidad se daría de forma continuada con los sistemas diseñados inmediatamente anteriores al sistema, permitiendo un rápido avance, nunca posible hasta ahora.

Entre los beneficios que proponen los expertos asistentes al workshop sobre el alcance y la naturaleza del PC (NRC, 2010), destacamos: conseguir una sociedad tecnológica exitosa, aumentar el interés por profesiones de las Tecnologías de la Información, facilitar el empoderamiento de las personas y apoyar la investigación de otras disciplinas aparentemente no relacionadas.

En relación a esto último, se puede apreciar una fuerte influencia del PC en otras disciplinas (NRC, 2010). Por ejemplo, la transformación que ha causado en la estadística gracias al gran tamaño de las muestras de datos, que permiten aumentar la fiabilidad de los estudios estadísticos.

Otro ejemplo de disciplina que se ha visto enriquecida gracias al pensamiento computacional es la biología. Gracias a la computación, se ha facilitado el reconocimiento de patrones a través de grandes cantidades de secuencias de datos.

Entre otras disciplinas en las que se puede observar el empleo del pensamiento computacional se incluyen la medicina y el cuidado de la salud, la investigación en

cáncer, las políticas públicas, la música, las leyes y el derecho, la economía y otras ciencias sociales, la física y la química (NRC, 2010).

Constable remarca en este workshop la importancia del doble sentido de la vía que conecta el pensamiento computacional con otras disciplinas. Él y otros expertos asistentes a este primer workshop sobre el PC (NRC, 2010) señalan que, sin la forma de pensar con la que trabajan otras disciplinas, habría sido difícil alcanzar el desarrollo actual con el que cuentan las Ciencias de la Computación y las Ciencias de la Información.

Enseñanza del pensamiento computacional

Como hemos visto, el pensamiento computacional es usado en muchas disciplinas sin una relación directa con las CC. Además, como sostiene Wing (2006, 2008, 2011), es un pensamiento necesario para todas las personas, ya que facilita la forma de enfrentarse a tareas diarias.

Como afirman Herde, Wüstenberg, y Greiff, (2016), todavía no se conocen los procesos cognitivos implicados en la resolución de problemas complejos pero las nuevas técnicas de análisis de datos están pronosticando poco a poco dichos procesos. Si conociésemos las estrategias que mejor pueden abordar la resolución de problemas complejos, podríamos destinar nuestros esfuerzos a enseñar dichas estrategias.

Si el llamado pensamiento computacional, que se traduce en una estrategia computacional, ayuda a resolver problemas complejos, reformulando éstos a términos más sencillos y manejables para nosotros, todas las personas deberían contar indudablemente con este pensamiento para hacer frente a los problemas y situaciones que se van a encontrar en la sociedad del siglo XXI.

Esta visión resalta la importancia de incorporar la enseñanza del pensamiento computacional en la educación formal, además desde edades tempranas si queremos que

todas las personas dominen en cierta medida el pensamiento computacional (Wing, 2008).

Hasta ahora se han visto diversos intentos de incluir el PC en la educación primaria, a lo que en Estados Unidos corresponde el K-12 (Valverde-Berrocoso, Fernández-Sánchez y Garrido-Arroyo, 2015). Entre los intentos más conocidos, encontramos la inclusión en asignaturas de programación y/o tecnología, como es el caso del Reino Unido y, aquí en España, la asignatura Tecnología, Programación y Robótica, implantada por la Comunidad de Madrid en el curso 2015/16.

Sin embargo, existen otros métodos de incluir el pensamiento computacional en la educación formal, que no necesariamente incluye la programación de código para conseguirlo. En este lado de la balanza encontramos un único caso, las escuelas Quest to Learn (Q2L, USA) que responden a la necesidad de los niños actuales, creando entornos de aprendizaje que responden al desarrollo integral (Valverde-Berrocoso et al., 2015).

Sin embargo, casi todos estos intentos dan por supuesto que, enseñando a programar a los niños, desarrollarán su PC, y eso es peligrosamente incierto. Si partimos del supuesto de que el aprendizaje de programación desarrolla procesos cognitivos del supuesto pensamiento computacional, lo lógico es pensar que enseñando a programar a los niños desarrollarán indiscutiblemente este pensamiento. Pero esta afirmación tiene que tomarse con cuidado, ya que no todos los cursos de programación desarrollan estrategias de pensamiento para programar.

Para que la programación sea un buen recurso para desarrollar el pensamiento computacional, necesitamos marcar ciertas directrices, ya que, como afirma Zapata-Ros (2015), en la actualidad la tendencia más frecuente es enseñar a programar de una forma poco reflexiva, enseñando a crear código desde las tareas más sencillas y lúdicas a las más complejas y aburridas.

Esta forma de enseñar programación tiene como objetivo formar a personas capaces de utilizar la programación en su futuro profesional, en lugar de tener como objetivo el desarrollar el pensamiento computacional. Esto conlleva a que, en muchas propuestas educativas, se aprenda a programar sin pensar sobre el problema a resolver, sin hacer un diseño previo a la creación del programa que permita reflexionar sobre cómo resolverlo. Como afirma Zapata-Ros (2015), fomentan un copia y pega de fragmentos de código para realizar operaciones concretas, sin entender la lógica que conlleva crear ese código.

Por estos motivos, creemos que es necesario proporcionar a largo plazo unas directrices adecuadas para incluir la enseñanza de este pensamiento, ya sea por medio de la enseñanza de la programación o por otros medios que impliquen poner en marcha los procesos cognitivos del pensamiento computacional.

Y todo esto es debido a que, aunque consideremos que en la sociedad en la que nos encontramos saber programar puede resultar útil para cualquier perfil profesional y no sólo para perfiles relacionados con las ciencias de la computación, la principal aportación que puede darnos el aprendizaje de la programación es utilizarla como una herramienta o recurso para desarrollar el pensamiento computacional.

Metodología empleada y lógica interna

Esta investigación emplea una metodología cuantitativa, empleando un estudio pre-post comparando los resultados entre un grupo que participa en una formación de 30 horas sobre programación y pensamiento computacional, y un grupo control que no participa en ninguna formación alternativa. De esta forma, podremos conocer el alcance del llamado pensamiento computacional, estudiando la relación entre el uso de las estrategias utilizadas por los científicos de la computación y la resolución de problemas complejos.

Articulación en capítulos: contenidos

Bloque 1. Marco teórico

Capítulo 1. Qué es el pensamiento computacional

En este capítulo se recoge toda la información relevante que se ha escrito hasta ahora del pensamiento computacional. En primer lugar, se presentan los primeros indicios del pensamiento computacional antes de ser considerado como tal. En segundo lugar, se exponen los enfoques que se han dado al pensamiento computacional, como herramienta cognitiva y como algo propio de las ciencias de la computación. A continuación, se recogen propuestas que intentan definir de forma operativa este pensamiento y que reclaman la necesidad de una definición operativa para poder progresar en la evaluación y el conocimiento del pensamiento computacional. Por último, se introduce la relación entre pensamiento computacional y resolución de problemas a partir de los procesos propuestos en la definición operativa del pensamiento computacional que se ha tomado como referencia (Computer Science Teacher Association y International Society Technology Education, 2011).

Capítulo 2. Resolución de problemas complejos

El capítulo dos se centra en recopilar el todo conocimiento relevante sobre la resolución de problemas, haciendo un recorrido desde el inicio de las investigaciones sobre este tema hasta la actualidad. Primero se presentan las ideas confirmadas a través de la investigación tradicional, en la que se diferenciaban claramente los estudios sobre la representación del problema y los estudios sobre el proceso para solución el problema, lo que permitió tener un mayor conocimiento sobre los factores que incluyen en cada parte. Posteriormente se introducen los problemas complejos, sus características y tipos, como una propuesta diferente a la hora de estudiar cómo se realiza la resolución de este tipo de problemas.

Capítulo 3. Aportaciones del pensamiento computacional a la RPC

En este capítulo se hace una reflexión sobre qué aporta el llamado pensamiento computacional a la resolución de problemas complejos (RPC), es decir, cómo mejora la resolución de problemas complejos gracias al empleo de este pensamiento. Para ello se profundiza en los procesos implicados en el pensamiento computacional, comprobando la relación entre estos procesos y los procesos implicados en la resolución de problemas.

Capítulo 4. Investigación sobre resolución de problemas complejos y pensamiento computacional

Este capítulo comienza resumiendo las técnicas tradicionales empleadas para la investigación sobre resolución de problemas. Se relata la evolución desde estas técnicas tradicionales de investigación hasta las corrientes actuales en la investigación en este campo. Posteriormente se recogen las investigaciones en resolución de problemas complejos o dinámicos, que adoptan el uso de marcos formales para mejorar sus cualidades psicométricas, además de explicar los problemas utilizados en estos marcos y otros problemas utilizados para evaluar la capacidad de resolución de problemas complejos. Por otra parte, en este capítulo se recogen las investigaciones que se han realizado sobre pensamiento computacional.

Bloque 2. Estudio empírico

Capítulo 5. Método de la investigación

En el capítulo cinco se recoge el método utilizado en el estudio llevado a cabo para comparar la relación entre pensamiento computacional y resolución de problemas complejos. Este capítulo tiene la estructura habitual de este tipo de capítulos: diseño de la investigación e hipótesis de estudio, participantes, instrumentos y materiales de investigación, y procedimiento empleado en el estudio.

Capítulo 6. Validación de las pruebas de RPC

En el sexto capítulo se realiza un análisis psicométrico de las pruebas diseñadas como parte del instrumento principal de evaluación para esta investigación. El estudio de fiabilidad se realiza con la prueba alfa de Cronbach para comprobar la consistencia interna de las pruebas, cuyo resultado debe ser al menos de 0,7 sobre 1. Con este resultado se puede afirmar que las pruebas de evaluación creadas miden de forma fiable la resolución de problemas complejos, siendo válidas para comprobar la hipótesis de la influencia del pensamiento computacional en la resolución de problemas. Este capítulo está en proceso de publicación, dado el impacto que puede tener la creación de este instrumento de evaluación validado para realizar estudios sobre la RPC en español.

Capítulo 7. Resultados y discusión

En este capítulo se recogen los resultados que muestran los diferentes análisis de datos realizados para comprobar las hipótesis de partida que nos marcamos como forma de explorar los datos, para comprobar la hipótesis general de la investigación acerca de la influencia del pensamiento computacional en la resolución de problemas. A partir de los datos obtenidos con estas pruebas, se realizan además análisis complementarios que nos permiten conocer más información sobre la relación entre nuestras dos variables.

Bloque 3. Conclusiones generales

Capítulo 8. Conclusiones y futuras líneas de investigación

En el último capítulo, el octavo capítulo, se establecen las conclusiones generales de nuestros resultados, que nos permiten establecer conclusiones generales sobre el tema de estudio. Además, estas conclusiones permiten establecer posibles líneas de investigación de cara a futuros estudios, basándonos en las cuestiones que no han quedado respondidas o que precisan de mayor evidencia empírica para ser confirmadas.

Bloque 1. Marco teórico

Capítulo 1. Qué es el pensamiento computacional

En este capítulo vamos a hacer una revisión teórica de los trabajos más relevantes sobre pensamiento computacional y las principales conclusiones extraídas a partir de estas investigaciones. El objetivo de esta revisión es tener una idea del estado actual de la cuestión, debido a que en estos últimos diez años se ha hablado mucho de la existencia de este pensamiento, de su importancia y de cómo desarrollarlo.

Qué se entiende por pensamiento computacional

El pensamiento computacional (PC) surge en 2006 desde el ámbito de las Ciencias de la Computación (CC). Jeannete Wing propone este pensamiento en las Comunicaciones de la ACM, Association for Computing Machinery de USA, como un conjunto de habilidades necesarias para resolver problemas complejos, aplicable universalmente y necesario para todas las personas, y no sólo para científicos de la computación (Wing, 2006).

Desde entonces, ha habido muchos intentos de definir la naturaleza de este pensamiento e incluso de consensuar unas directrices pedagógicas para desarrollarlo (National Research Council, 2010; 2011).

Sin embargo, casi todos los intentos se han producido desde el ámbito de las CC y desde la educación en tecnología, sin una aproximación psicológica que defina cómo operan los procesos cognitivos implicados en este supuesto pensamiento y para qué nos resulta útil a las personas utilizar un pensamiento computacional.

Por este motivo, el pensamiento computacional todavía no ha recibido una definición consensuada, sino que existen deficiencias en las características conceptuales (Weinberg, 2013). Como afirma Hu (2011), si no conseguimos establecer una definición

precisa del PC, nunca podrá considerarse importante porque no seremos capaces de evaluarlo y saber cómo mejorarlo.

Para intentar resolver estas cuestiones ya en los años 2010 y 2011, se realizaron dos workshops para tratar estos temas donde expertos que habían realizado estudios sobre el pensamiento computacional pudieran compartir sus hallazgos y avances en el campo y llegar a un consenso entre todos los expertos asistentes a dichos workshops.

De estos eventos se extrajeron conclusiones interesantes, recogidas en reports, el primero de ellos llamado Report of a Workshop on the Scope and Nature of Computational Thinking (National Research Council, 2010), y el segundo de ellos llamado Report of a Workshop of Pedagogical Aspects of Computational Thinking (National Research Council, 2011).

En particular, en el primer workshop sobre el alcance y la naturaleza del pensamiento computacional, se plantearon cinco cuestiones importantes aún sin respuesta, y que después de varios años, siguen sin una respuesta ampliamente aceptada y definitiva. Estas cuestiones siguen aún abiertas, a la espera de una respuesta, y son:

1. ¿Cuál es la estructura del pensamiento computacional?
2. ¿Cómo se puede reconocer a alguien que piensa computacionalmente?
3. ¿Cuál es la conexión entre tecnología y pensamiento computacional?
4. ¿Cuál es la mejor pedagogía para promover el pensamiento computacional?
5. ¿Cuál es el papel institucional adecuado de la comunidad de Ciencias de la Computación con respecto al pensamiento computacional?

Durante esta investigación se intentará avanzar en algunas de estas cuestiones, o al menos adoptar el enfoque que consideremos más adecuado, de cara a una mejor comprensión de lo que puede abarcar el pensamiento computacional.

Primeros indicios de este pensamiento

Décadas antes de que Wing acuñase el término de pensamiento computacional aparecieron las primeras argumentaciones desde las CC, a favor de la existencia de unas habilidades propias de este campo y útiles en general, aunque no hablaran de pensamiento computacional como tal (National Research Council, 2010).

Ya en los años 70, Seymour Papert subrayó los beneficios en el desarrollo si los niños aprendían a programar desde edades tempranas (Papert, 1972). Entre las ideas básicas que recoge en su publicación, destaca la idea de que la programación puede ayudarles a articular el trabajo de su mente, particularmente la interacción entre el niño y la realidad en el transcurso del aprendizaje y el pensamiento.

Papert es popularmente conocido por su programa LOGO, que consiste en una aplicación para aprender a programar que además ayuda a aprender matemáticas, concretamente geometría (National Research Council, 2010; Valverde-Berrocoso, Fernández-Sánchez y Garrido-Arroyo, 2015).

LOGO es el antecesor de todos los programas creados recientemente para enseñar programación a niños, como Scratch o Alice (National Research Council, 2010), programas con una interfaz más atractiva y con una sintaxis simplificada para facilitar el aprendizaje de la lógica básica de programación.

Además de Papert, en los últimos 25 años se han publicado informes de asociaciones estadounidenses relacionadas con las Ciencias de la Computación que reclaman que determinadas habilidades y técnicas de pensamiento propias de los científicos de la computación, facilitan el pensamiento.

Entre los informes publicados en la década de los 90, destacan el *Computing the Future* (Hartmanis, y Lin, 1992) publicado por el Consejo de Investigación Nacional de

Estados Unidos, y el *FITness report o Being Fluent with Information Technology* (Snyder, Aho, Linn, Packer, Tucker, Ullman, y Van Dam, 1999), publicado por la Junta de Ciencias de la Computación y Telecomunicaciones del Consejo de Investigación Nacional de Estados Unidos.

En el primero de estos informes, *Computing the Future* (Hartmanis y Lin, 1992), se reclama las Ciencias de la Computación como una disciplina intelectual. En este caso concreto, el término ciencia se refiere a entender actividades computacionales, a través de modelos matemáticos y de ingeniería basados en teoría y abstracción.

Por tanto, la clave intelectual reside en el pensamiento algorítmico, la representación de información y los programas de ordenador. Las cuestiones que relacionan la clave intelectual de esta disciplina con el pensamiento computacional son de dos tipos: por una parte, cómo la abstracción debe ser representada y, por otra parte, cómo debería ser la estructura de la representación para que las operaciones comunes tengan un acceso eficiente (Hartmanis y Lin, 1992).

En el segundo de los informes, *FITness report* (Snyder, et al., 1999), se pretende formular la base del conocimiento de las Ciencias de la Computación identificando tres categorías, consideradas igualmente importantes. Estas categorías son: capacidades cognitivas/intelectuales, conceptos computacionales y habilidades para las tecnologías de la información.

Mientras que la primera, capacidades cognitivas tiene que ver con el razonamiento lógico y la resolución de problemas, como depurar errores; la segunda, conceptos computacionales se relaciona con ideas que surgen de la tecnología, como la programación y los algoritmos (Snyder, et al., 1999, p. 2).

Entre el año 2000 y el 2010, podemos destacar tres informes que causaron impacto de cara al surgimiento del pensamiento computacional. Estos informes, ordenados por fecha de publicación son: *el Technically Speaking Why All Americans Need to Know More About Technology* (Pearson y Young, 2002) publicado por la Academia Nacional de Ingeniería y el Consejo de Investigación Nacional de USA; el informe *Computer Science: Reflections on the Field* (Shaw, 2004) por el Consejo de Investigación Nacional de USA; y el *Engineering in K-12 Education: Understanding the Status and Improving the Prospects* (Teacher Advisory Council, 2009), publicado también por la Academia Nacional de Ingeniería y el Consejo de Investigación Nacional de Estados Unidos. Este último informe fue publicado después de que surgiera el concepto de pensamiento computacional (Wing,2006).

En el primero de estos informes, *Technically Speaking Why All Americans Need to Know More About Technology* (Pearson y Young, 2002), se defiende la existencia de tres dimensiones en la alfabetización tecnológica, dimensiones que guardan cierta correspondencia con las categorías ya propuestas en el *FITness report* (Snyder,et al., 1999). Estas dimensiones son el conocimiento para la alfabetización tecnológica, los caminos de pensamiento y algunas capacidades técnicas básicas en la alfabetización tecnológica.

En el segundo de estos informes, *Computer Science: Reflections on the Field* (Shaw, 2004), podemos destacar el ensayo El legado de Ciencias de la Computación, escrito por Gerald Sussman, quien señala que las Ciencias de la Computación no es una ciencia en sí misma, sino que más bien se trata de una revolución en la forma de pensar y de comunicar ideas y métodos.

El último de los informes, *Engineering in K-12 Education: Understanding the Status and Improving the Prospects* (Teacher Advisory Council, 2009), resalta la ausencia

de la ingeniería en los proyectos STEM dirigidos al K-12, lo que en España correspondería a Educación Primaria. Esta ausencia se debe en parte a la falta de unos estándares adecuados para la enseñanza, la monitorización y la evaluación de la ingeniería. Además, tampoco se ha invertido lo suficiente en la formación y el desarrollo de los profesionales de la educación para esta etapa educativa.

Para mejorar la educación en ingeniería, el informe propone incorporar el diseño en ingeniería; enfatizar habilidades importantes entorno al conocimiento de las matemáticas, la ciencia y la tecnología, lo que serían métodos computacionales; y promover hábitos mentales ingenieriles, es decir, habilidades de pensamiento asociadas a la ingeniería (Teacher Advisory Council, 2009).

En la revisión que hemos realizado, podemos observar que los aspectos del pensamiento computacional se vienen planteando desde años atrás de que surgiera este concepto.

Desde hace tiempo se ha visto cómo los aspectos clave de la computación y del pensamiento computacional han influido en otras disciplinas, permitiendo un mayor avance de éstas (Wing, 2014). De igual forma, diferentes disciplinas han aportado a las Ciencias de la Computación métodos y técnicas que han permitido avanzar y evolucionar la disciplina (National Research Council, 2010).

Desde que surgió el concepto de pensamiento computacional, también ha habido intentos de formalización y unificación de la definición del pensamiento computacional, qué es y qué no es, qué implica y qué procesos cognitivos incluye este pensamiento. Algunas instituciones están teniendo iniciativas para fomentar el aprendizaje de las CC y el pensamiento computacional, y formar a los educadores implicados (National Research Council, 2010).

Sin embargo, aunque hay esfuerzos por avanzar en la investigación sobre el pensamiento computacional, no se ha consensuado aún cuáles son los procesos implicados en el pensamiento computacional ni en qué aspectos pueden resultarnos útiles.

Intentos de una definición consensuada

Para poder crear una definición adecuada del pensamiento computacional, debemos empezar por definir qué es el pensamiento.

Definir el pensamiento de forma concisa y adecuada es una tarea compleja que todavía no ha recibido una respuesta convincente. Carretero y Asensio (2004) proponen una aproximación genérica del concepto:

Pensamiento designa lo que contiene o aquello a lo que apunta un conjunto de actividades mentales u operaciones intelectuales, como razonar, hacer abstracciones, generalizar, etc., cuyas finalidades son, entre otras, resolver problemas, tomar decisiones y representarse la realidad externa.

(p.14)

Dentro del estudio del pensamiento, se distinguen varios *subcampos* de estudio como son el razonamiento, la toma de decisiones o la resolución de problemas (RP) (Carretero y Asensio, 2004; Holyoak y Morrison, 2012).

Como afirman estos autores, el campo de estudio de la resolución de problemas se realiza desde una perspectiva diferente a otros subcampos del estudio del pensamiento, pero todas estas perspectivas están relacionadas, es decir, existe una interrelación entre los subcampos de estudio del pensamiento. En este sentido podemos entender la resolución de problemas como una toma de decisiones que implica razonamiento.

Wing (2006) señala que el pensamiento computacional es útil para la resolución de problemas complejos, y de ahí se extrae nuestro objetivo.

El objetivo general de esta investigación consiste en conocer cómo puede facilitar la resolución de un problema el empleo de un pensamiento computacional, es decir, queremos comprobar la aportación del pensamiento computacional a nuestro conocimiento sobre la resolución de problemas.

Para ello, en este primer capítulo vamos a recapitular todo lo que se ha escrito en estos últimos años sobre el pensamiento computacional y vamos a exponer brevemente la relación entre ambos, profundizando en su comparación durante el tercero de los capítulos, en el que comparamos los procesos implicados.

Wing (2006, 2008), en su propuesta para acuñar el término pensamiento computacional, lo define como un camino de cómo los humanos resuelven problemas y diseñan sistemas que no seríamos capaces de hacerle frente de forma individual. Según esta autora, este pensamiento implica reformular un problema aparentemente difícil en uno que sabemos resolver, ya sea por reducción, incrustación, transformación o simulación (Wing, 2006, p. 33).

Años después, esta misma autora intenta definir el PC de una forma más concreta y descriptiva. Junto a otros autores, Cuny, Snyder y Aho, consiguen una definición más precisa del término, el cual queda definido como un conjunto de procesos de pensamiento implicados, una actividad mental para formular problemas de forma que admitan una solución computacional (Cuny, Snyder y Wing, NP; citado en Wing, 2011, 2014).

Según Wing (2011) una solución computacional es más que una solución en un problema matemáticamente bien definido, de forma que la solución es completamente analizable, sino que también incluye las soluciones aplicadas a sistemas complejos.

Finalmente, Wing (2014) incluye un aspecto más en el pensamiento computacional, y es que este pensamiento no está solamente relacionado con el proceso de resolución de problemas, sino que también se relaciona con la formulación de éstos.

A partir de este aspecto que incluye Wing en 2014, surge nuestro objetivo que consiste en comprobar si el pensamiento computacional beneficia al proceso de resolución de problemas, a la formulación de éstos, entendiendo formulación como la representación de los mismos, o a ambos.

Por otra parte, en el workshop organizado en 2010 para concretar el alcance y la naturaleza del pensamiento computacional, podemos apreciar diferentes puntos de vista diferentes sobre cuál es el aspecto clave de este pensamiento. A continuación, recogemos los aspectos que nos parece más interesante destacar.

PC como herramienta cognitiva

Entre las principales perspectivas recogidas, destaca la aproximación a este concepto definiéndolo como determinados procesos cognitivos necesarios para ser aplicados en cualquier ciencia. Wulf defiende que estos procesos se centran en la abstracción para hacer posibles otros procesos, concretamente el término computacional se centraría en los números (National Research Council, 2010, p.11).

Wing y Sussman afirman que el PC puede ser visto como un puente entre la ciencia y la ingeniería, como una meta-ciencia sobre el estudio de los caminos o métodos de pensamiento que son aplicables a través de diferentes disciplinas. Sussman destaca además que el PC es un camino de formular métodos precisos para hacer cosas, para llevar a cabo procedimientos y análisis de forma rigurosa (NRC, 2010, p.12). De estas afirmaciones podemos extraer que el PC sería un pensamiento necesario para aplicar el método científico de forma adecuada y meticulosa.

Sin embargo, algunos asistentes como David Moursund, discuten que el llamado “pensamiento computacional” es demasiado similar al pensamiento procedimental que detalla Seymour Papert en *Mindstorms*. Según este autor, el pensamiento procedimental incluye procedimientos de desarrollo, representación, testeo y depuración; y un procedimiento paso a paso efectivo, suficientemente detallado como para que se pueda llevar a cabo por un agente específico de forma mecánica (NRC, 2010, p.11).

Relación entre PC y las ciencias de la computación

Los intentos de definición del PC no quedan satisfechos en este encuentro debido al enfrentamiento existente entre los expertos que consideran que el PC incluye la capacidad de decidir cómo resolver un problema con la tecnología existente, que corresponde a una de las preguntas que quedan abiertas en este workshop (NRC, 2010). A continuación, revisamos ambas posturas y destacamos los aspectos clave de ambas visiones del PC.

Por una parte, algunos asistentes consideran que el pensamiento computacional es un conjunto de procesos cognitivos que no implican el uso de la tecnología. Esta perspectiva considera que el único requisito es un agente que computa, es decir, que procesa un conjunto de herramientas mentales para hacer frente a problemas computacionales, es decir, problemas que se pueden resolver de forma computacional (NRC, 2010, p.26).

Por ejemplo, Peter Lee defiende que el pensamiento computacional es fundamental para expandir las capacidades mentales a través de herramientas abstractas que ayudan a manejar la complejidad, además de permitir la automatización de tareas. Su visión no solo se reduce a la forma de replicar estos procesos en la inteligencia artificial en las Ciencias de la Computación, sino que tiene una orientación más amplia que abarca

los mecanismos de inteligencia humana que pueden producir aplicaciones prácticas (NRC, 2010, p11).

En el encuentro, Marcia Linn, Joshua Danish y Tim Bell presentan ejemplos de formas de desarrollar el pensamiento computacional sin usar ordenadores, en proyectos como el *Computer Science Unplugged*. Estos ejemplos forman parte de una visión compartida por los asistentes que ven el PC como una herramienta para hacer frente a los problemas de cada día (NRC, 2010, p.20).

Por otra parte, el otro enfoque que defienden algunos de los expertos, es que el PC implica conocer y utilizar la tecnología más apropiada para resolver un problema. Desde este punto de vista, los ordenadores y otros dispositivos computacionales son los que permiten lograr el PC (NRC, 2010, p.26). Entre quienes defienden esta postura, encontramos a Andrew McGettrick que destaca el constante esfuerzo de mantenerse actualizado en la tecnología (NRC, 2010, p11).

En este sentido, Moursund afirma que el PC fusiona los conceptos *affordance*² y *persona*, a las tecnologías de la información y los ordenadores (NRC, 2010, p18). Según esta postura, quienes conozcan sobre más número de tecnologías diversas, serán quienes tengan más posibilidades para resolver problemas.

En la misma línea, Robert Constable alega que lo relevante del pensamiento computacional es precisamente que los ordenadores puedan ejecutar nuestros pensamientos computacionales y que los ordenadores empiezan a ser nuestros compañeros y colaboradores en nuestros descubrimientos (NRC, 2010, p12).

² El término *affordance* fue acuñado por Gibson en 1977 para describir las posibilidades de acción que ofrece una situación, es decir, según los datos que recopilamos de la situación, pensar en posibles soluciones familiares para la persona.

Dentro de este segundo enfoque, podemos diferenciar una segunda visión en la relación de PC y tecnología. Algunos asistentes como Roy Pea y Ursula Wolz, comparten la visión de que el PC tiene la base en la programación, teniendo una visión más concreta que quienes se refieren a conocer la tecnología existente. Mitchel Resnick añade a esta visión que la programación es un significado de expresión y ayuda a desarrollar nuevos caminos de pensamiento. Andrea diSessa reclama la enseñanza del pensamiento computacional como una nueva forma de alfabetización, la alfabetización computacional, desde el punto de vista de aprender a programar (NRC, 2010, p.13).

Con estos argumentos, muchos de los expertos reunidos en dicho encuentro consideran que, en la sociedad actual, el PC es igual de importante que otras habilidades fundamentales como la aritmética y el lenguaje verbal, ya sea aplicado en la lectura, la escritura o la comunicación oral (NRC, 2010, p.13).

En este sentido, Alan Kay, rechaza la visión del PC como lenguaje, haciendo una comparación entre el lenguaje verbal, universal e innato ya que es compartido por todos los humanos; y el lenguaje escrito, la ciencia o las matemáticas deductivas, los cuales se desarrollan sólo en las culturas que se demandan, por lo que no podemos decir que sean universales e innatos. El PC estaría en este segundo caso, ya que no es necesario en todas las sociedad o culturas, por lo que no podemos considerar que sea un lenguaje como tal (NRC, 2010, p.16).

Como vemos, esta segunda postura acerca del PC y la tecnología incluye dos visiones:

- (1) el PC incluye la aplicación de las *affordance* surgidas al conocer la tecnología actual;
- (2) el PC requiere una alfabetización tecnológica, el saber crear programas tecnológicos, y los procesos cognitivos implicados en programación

implican supuestamente el desarrollo del PC, incluso de una alfabetización computacional.

Por último, hay que saber diferenciar el pensamiento computacional de otros conceptos relacionados (NRC, 2010, p. 28). Los autores que aparecen a continuación, entre otros, expusieron las siguientes diferencias:

- Constable lo diferencia de la concepción tradicional de *alfabetización digital*, en la que se consideraba que lo fundamental era aprender a manejar software concretos, como hojas de cálculo o procesadores de textos.
- Sussman también hace hincapié en la diferencia existente entre PC y *Ciencias de la Computación*, insistiendo en que el PC es parte de las CC, como el pensamiento matemático es parte de la disciplina de las matemáticas pero no se reduce a ella.
- Snyder se centra en diferenciar el PC de lo que originalmente se planteó como *fluidez con las tecnologías de la información* (Snyder et al.,1999). Snyder cree que en la actualidad se podría reformular para incluir el pensamiento computacional como parte de lo que se considera necesario para conseguir esa fluidez. Sin embargo, otros expertos creen que ambos conceptos son cualitativamente diferentes.

Otros procesos inspirados en las Ciencias de la Computación

Se han descrito otros procesos que supuestamente llevamos a cabo con el pensamiento computacional, procesos inspirados en las Ciencias de la Computación. Sin embargo, entre los procesos encontrados no hemos podido considerar ninguno de ellos como un proceso cognitivo implicado en el PC, ya fuese por no tratarse de un proceso cognitivo en sí, por ser parte de otro proceso cognitivo ya incluido o por no ser lo suficientemente concreto como para incluirlo en nuestra definición operativa.

Entre estos procesos podemos nombrar la recursividad (Wing, 2006); pensar en términos de prevención en la experiencia de usuario, anticipándose a los usuarios y poniéndose en el peor caso (Wing, 2006); *testing* o pruebas, *debugging* o depuración de código, optimización del código o de los procesos, etc. (NRC, 2010).

Aunque se habla continuamente de términos de las Ciencias de la Computación, estos conceptos no son únicos y exclusivos de las CC, sino que se comparten con otras ciencias e ingenierías (NRC, 2010).

Procesos que se solapan con otros tipos de pensamiento

Entre los procesos y otros tipos de pensamiento que más se citan en la literatura original referente al PC (Wing, 2006, 2008, 2011), encontramos:

- El pensamiento recursivo, relacionado con la recursividad.
- El pensamiento paralelo, relacionado con el paralelismo.
- El pensamiento analítico, relacionado con el análisis de datos.
- El pensamiento algorítmico, relacionado con la creación de algoritmos.
- El pensamiento matemático, relacionado con la forma en la que se da la RP.
- El pensamiento ingenieril, relacionado con la forma en la que diseñamos y evaluamos un sistema complejo que opera con las restricciones del mundo real.
- El pensamiento científico, relacionado con la forma en la que entendemos un problema que nos permite computarlo o dividirlo en pequeños cómputos.
- El pensamiento procedimental, que comparte procesos muy similares al PC.

Además de citar otros tipos de pensamiento y otros procesos cognitivos como el pensamiento lógico, el pensamiento de sistemas, el reconocimiento de patrones, el razonamiento compositivo, etc. (Wing, 2011).

Definición operativa del PC

Si resulta difícil consensuar una postura sobre qué implica a grandes rasgos el pensamiento computacional, no podía ser menos complicado consensuar los procesos cognitivos que éste implica. En la literatura revisada, podemos apreciar diferentes enfoques del pensamiento computacional en los que se repiten a menudo determinados procesos como base del pensamiento computacional.

Desde los inicios de la propuesta de este pensamiento en 2006, se han propuesto diferentes definiciones que recogen determinados procesos de forma más o menos repetida a lo largo de las diferentes propuestas. Cabe señalar tres propuestas que, en nuestra opinión, han aportado mucho al camino seguido hacia una definición consensuada y única del pensamiento computacional. El resto de propuestas de la literatura se recogen en el capítulo cuatro, en el que se cuestionan dichas propuestas por motivos relacionados con la forma de operativizar el PC y abordar su evaluación.

La primera de ellas, propuesta por el Computer Science Teacher Association y el International Society Technology Education (CSTA y ISTE, 2011), recoge de forma operativa las características que según ellos componen el pensamiento computacional.

La segunda propuesta, por Selby y Woollard (2013), pretende establecer una definición a partir de una revisión de todas las publicaciones relevantes existentes hasta la fecha de su publicación, incluyendo y excluyendo las características y/o procesos cognitivos, con mayor y menor aparición en la literatura, respectivamente.

La tercera y última de las propuestas recogidas, de Zapata-Ros (2015) la hemos querido destacar por las conclusiones que arroja este autor sobre los intentos de definición de los procesos encontrados.

CSTA y ISTE (2011)

En 2011, la asociación de profesores de Ciencias de la Computación (*Computer Science Teachers Association*, CSTA) y la Sociedad Internacional de Tecnologías en Educación (*International Society for Technology in Education*, ISTE), intentaron recoger, y unificar en una definición operativa, los diferentes procesos o características que definen el pensamiento computacional.

El CSTA y el ISTE (2011) proponen la primera definición operativa del pensamiento computacional, basándose en las características o procesos recogidos hasta la fecha. Estas características son abstracción, descomposición del problema, recopilación de datos, análisis de datos y representación de datos, secuenciación o algoritmos, automatización, paralelismo y simulación (CSTA y ISTE, 2011).

El marco conceptual formado por el conjunto de estas características, representa con bastante acierto lo que se ha venido estudiando hasta la fecha sobre qué es y qué implica el PC, por lo que continúan resultando de gran utilidad para el estudio del PC. En esta revisión vamos a incluir todo lo nombrado sobre cada una de estas características en la historia de la investigación sobre pensamiento computacional, y no únicamente lo que se recoge en la definición operativa de CSTA e ISTE (2011).

Abstracción

Así, el término más repetido a través de estas propuestas es *abstracción*. La abstracción es un proceso cognitivo que consiste en desprender una situación de sus particularidades, de modo que nos quedemos únicamente con las propiedades generales (Wing, 2006, Wing 2008, Wing 2010). De esta forma, podremos comparar los aspectos relevantes de la situación con otras situaciones similares.

En palabras de Wing (2006, 2008, 2010) las abstracciones son las herramientas mentales de la computación y, por tanto, son la esencia del pensamiento computacional. Dicho de otro modo, en la computación necesitamos trabajar con múltiples capas y/o niveles de abstracción y debemos ser capaces de entender las relaciones existentes entre las diferentes capas.

Descomposición del problema

La descomposición también es considerada una de las claves del pensamiento computacional. Cuando nos enfrentamos a una tarea larga y compleja, o cuando tenemos que diseñar un gran sistema complejo necesitamos descomponerlo en pequeñas partes (Wing, 2006).

Esta tarea de descomposición incluye la reformulación del gran problema para poder abordar de una forma más sencilla (NRC, 2010). La reformulación puede hacerse, principalmente, por reducción o transformación (Wing, 2006; NRC, 2010).

Tratamiento de datos: recopilación, análisis y representación de datos

Wing (2008) se refiere a la mejora que produce el PC en el tratamiento de datos cuando afirma que el pensamiento computacional profundo ayuda, no sólo a modelar sistemas cada vez más complejos, sino que nos facilita el análisis de grandes cantidades de datos que recopilamos y generamos (p. 3720).

A partir de esta definición, se han recogido otras que mantienen una aproximación similar de cada proceso relacionado con el tratamiento de datos (NRC, 2010).

Con recopilación de datos se refieren principalmente a la constante búsqueda de nuevos datos y el uso de cantidades masivas de éstos.

El análisis de datos se puede interpretar de varias formas, ya que dependerá si nos referimos al contexto del problema, en el que se relaciona el análisis con la abstracción y

la descomposición del problema; o si nos referimos al análisis de los datos de la solución, en el que se relaciona principalmente con evaluación (Selby y Woollard, 2013).

En las primeras de sus interpretaciones, el análisis de datos conduce a una generalización o representación del problema (Wing, 2006). Para poder alcanzar una generalización a partir de los datos analizados, el PC requiere de un análisis dimensional que permita comprobar los tipos y modelos con los que trabajamos.

Por otra parte, Wing (2006, 2008) ve el pensamiento computacional como un tipo de pensamiento analítico. Define el PC como planificar y programar.

El supuesto aspecto del PC que sugiere Moursund (NRC, 2010), quien defiende la necesidad de dar respuesta a los problemas a través de la tecnología, también requeriría de una capacidad analítica para reconocer las *affordance* surgidas entre el problema y la tecnología disponible.

También, relacionada con la descomposición del problema y su resolución en pequeñas partes, se requiere el análisis de cada una de las pequeñas partes de forma independiente para reconocer modelos situacionales similares para aplicar las soluciones ejecutadas en las situaciones pasadas (Selby y Woollard, 2013).

Por último, cuando estas descripciones del PC citan la representación de datos (NRC, 2010), normalmente se refieren a crear una correcta representación del problema, ya que de esto dependerá el crear una representación acertada de la estructura más eficiente para analizar los datos.

Como vemos, según las ideas comunes a algunas de las definiciones revisadas, los procesos de análisis y representación de datos se relacionarían de forma bidireccional. Por una parte, como acabamos de mencionar, el análisis de los datos conduce a una generalización o representación del problema. Por otra parte, para asegurar una

representación adecuada, necesitamos un continuo análisis de los datos que vamos generando con la representación creada.

Algoritmos

Aunque en la literatura revisada no se hace especial hincapié en este aspecto del pensamiento computacional, es una pieza clave y necesaria en el PC.

Wing (2010) señala que el pensamiento computacional incluye pensamiento algorítmico. Define un algoritmo como una abstracción de un proceso que recibe datos, ejecuta la secuencia de éstos como una lista de pasos y genera datos para satisfacer el objetivo deseado (p.1).

Entre los expertos del pensamiento computacional se considera el análisis de la ejecución del algoritmo como una parte fundamental del análisis de los datos y los procesos cognitivos llevados a cabo (NRC, 2010).

Automatización

Las pequeñas tareas que no requieren apenas toma de decisiones se pueden automatizar, con el fin de ahorrar en tiempo y esfuerzo de cara a la resolución de un problema que implique pensamiento computacional. De esta forma, y relacionado con el siguiente proceso descrito, las tareas automáticas se pueden paralelizar, lo que nos lleva a disminuir aún más el tiempo y esfuerzo requerido.

En este punto, podría surgir el conflicto en el lector de si el pensamiento computacional depende de la tecnología actual para poder desempeñarse. La mayor parte de tareas que se automatizan en la actualidad, son realizadas por máquinas tecnológicas, lo que nos hace pensar que sí es necesario hablar de tecnología para hablar de automatización.

Sin embargo, nosotros creemos que las tareas se pueden realizar de forma automática sin necesidad de emplear una máquina. Por ejemplo, el físico Richard P. Feynman desarrolló en la Segunda Guerra Mundial un sistema automático, que usaba personas en lugar de máquinas, un sistema de “computación paralela” en la que cada una se especializaba en una parte del proceso, de forma que consiguió automatizar todos los procesos que debía llevar a cabo (Villatoro, 2012).

Paralelismo

El concepto de paralelismo, en la definición operativa de pensamiento computacional, se refiere principalmente al paralelismo entre acciones. Los lenguajes de programación trabajan con este principio, al realizar secuencias de instrucciones que suceden de forma simultánea (Brennan y Resnick, 2012).

Wing (2006, p.1) afirma que el PC implica realizar un procesamiento paralelo, algo que los expertos en PC también incluyen en la definición que proponen para el pensamiento computacional (NRC, 2010). Sin embargo, en la literatura revisada no se ha visto ninguna descripción sobre cómo actúa supuestamente este procesamiento paralelo en el pensamiento computacional.

Simulación

La simulación en el pensamiento computacional es importante cuando la relacionamos con los complicados procesos de abstracción que se necesitan en el PC. En este sentido, nosotros debemos tener en mente la relación entre cada par de capas, definiendo esta relación a través de una función de abstracción (Wing, 2008). Para ello, necesitamos trabajar con *simulación* y crear diferentes tipos de asignaciones. Las asignaciones creadas nos sirven para mostrar la equivalencia entre una máquina de estados abstractos y uno de sus posibles estados a refinar.

Selby y Woollard (2013)

Selby y Woollard (2013) consideran que la unificación de los procesos cognitivos del PC ayudará a su inclusión en los currículos educativos, ya que hasta ahora estos currículos parecen una colección de los recursos que les interesan a determinados expertos en educación, dependiendo de las influencias con las que cuente la persona que diseña el currículo educativo.

Por ello, estos autores intentan construir una definición operativa del pensamiento computacional, basándose en los trabajos realizados hasta la fecha de publicación del artículo. Estos autores consideran, junto a otros muchos, que definir operativamente el PC nos ayudará a depurar todas las aproximaciones existentes y buscar el consenso entre los expertos que estudian el pensamiento computacional.

Además, señalan como beneficios que crear esta definición nos ayudará a facilitar su evaluación y comprobar la validez de los trabajos realizados hasta la fecha, y también nos permitirá crear actividades de aprendizaje más efectivas para desarrollar el pensamiento computacional (Selby y Woollard, 2013, p.1).

Para crear esta definición buscaron entre los artículos analizados los términos que se repetían en las descripciones del concepto. Después, analizaban las equivalencias entre términos para agrupar los términos similares. Estos autores comprobaron el significado de los términos en cada artículo, ya que incluso refiriéndose a los mismos términos, no en todos los artículos revisados significaban lo mismo.

A partir de este análisis, Selby y Woollard (2013) proponen una definición operativa del PC: el pensamiento computacional es una actividad, a menudo orientada a un producto, asociada con la resolución de problemas, pero no limitada a ésta (p.5). En otras palabras, es un proceso cognitivo centrado en la RP que incorpora:

- La habilidad para pensar en abstracciones
- La habilidad para pensar en términos de descomposición
- La habilidad para pensar de forma algorítmica
- La habilidad para pensar en términos de evaluación
- La habilidad para pensar en generalizaciones

Como vemos en esta definición, los procesos centrales implicados son prácticamente los mismos que proponen la CSTA y la ISTE (2011) en su definición operativa. Por tanto, con la propuesta de Selby y Woollard (2013) se confirma que los procesos que recogieron las asociaciones CSTA e ISTE se han propuesto de forma acertada, ya que son los procesos que más se han estudiado a través de las diferentes investigaciones que se han hecho en relación con este pensamiento.

Zapata-Ros (2015)

Zapata-Ros (2015) intenta crear una lista de los componentes relativos al pensamiento computacional. Sin embargo, esta lista de componentes, al igual que la mayoría de los intentos de definir operativamente el pensamiento computacional, no han permitido establecer aún unos procesos totalmente delimitados. En las definiciones revisadas hasta ahora, cada autor propone su propia lista de procesos indicando lo que para él significa el pensamiento computacional.

Si analizamos los componentes propuestos por los diferentes autores, vemos que ni siquiera existe una definición apropiada para cada componente o proceso. Esta limitación provoca que algunos componentes se refieran a diferentes significados en cada ocasión; además de no ser excluyentes entre sí, es decir, que varios componentes recogidos pueden hacer referencia a procesos cognitivos similares (Zapata-Ros, 2015).

Este autor (Zapata-Ros, 2015) pone de manifiesto que esta cuestión complica en gran medida el poder crear una taxonomía o lista de elementos que puedan ser aceptados como elementos del PC. Añadiendo la dificultad creada por las diferentes escalas, o niveles conceptuales, en las que se encuentran los elementos listados en estas propuestas operativas.

Los argumentos expuestos hasta ahora son suficientes para considerar necesario el empleo de una definición operativa, más o menos consensuada y definitiva, que permita conocer los procesos cognitivos que implica el pensamiento computacional, de cara a mejorar el estudio del PC, ya que facilita la creación de intervenciones educativas que aseguren el desarrollo de este pensamiento, de cara a conocer cómo el pensamiento computacional aporta valor si se emplea en la resolución de problemas complejos.

Relación entre pensamiento computacional y resolución de problemas

Para hacer una introducción sobre las similitudes de la resolución de problemas y el pensamiento computacional, antes de profundizar en esta relación en el tercer capítulo, vamos a describir brevemente en qué aspectos podemos suponer que son similares y en qué aspectos el pensamiento computacional aporta un elemento distintivo que permite enfocar de forma diferente la resolución de problemas, a partir de la literatura recogida sobre el tema.

Para nuestra investigación, nos vamos a basar en el enfoque de sistemas complejos múltiples para la evaluación de la capacidad de resolución de problemas complejos (Greiff, Wüstenberg y Funke, 2012), concretamente en el marco establecido por estos autores para las pruebas del programa internacional de evaluación de estudiantes PISA 2012 (OECD, 2010) que se explicará más adelante, ya que en este enfoque se descompone el proceso de RP en diferentes subprocesos que forman parte de la RP.

Este marco es perfecto para nuestra investigación, ya que está enfocado para su empleo con personas de 15 años, al igual que nuestros participantes. Además, se basa en el enfoque de sistemas múltiples complejos (Greiff, Wüstenberg y Funke, 2012) que permite una evaluación fiable de la resolución de problemas complejos.

Este marco de subprocessos engloba prácticamente todo el proceso de RP y permite medir cada subprocesso de forma adecuada e independiente, permitiéndonos un mayor conocimiento de cómo afecta cada aspecto estudiado a cada parte del proceso. Estos procesos son exploración y entendimiento, representación y formulación, planificación y ejecución, y monitorización y reflexión.

Comparar estos procesos con los procesos descritos en el pensamiento computacional es posible si realizamos equivalencias entre los términos empleados, ya que estos términos diferentes entre sí, implican procesos similares.

Para ello, nosotros vamos a utilizar el marco conceptual configurado por la CSTA y la ISTE (2011) que proporcionan una serie de procesos que a menudo se ven en las diferentes definiciones de pensamiento computacional y que en conjunto forman una definición bastante satisfactoria del concepto.

El proceso de exploración equivale al proceso de recopilación. El proceso de entendimiento implicaría los procesos de abstracción y análisis, ya que la abstracción debe extraer la idea o modelo mental adecuado, y el análisis de esta idea debe confirmar que la abstracción se ha realizado correctamente.

El proceso de representación equivale, obviamente, al proceso de representación en pensamiento computacional. Aunque el primero se define a la representación del problema en sí, y el segundo hace alusión a la representación de datos; este último en realidad se refiere a la representación de los datos del problema.

El proceso de formulación, relacionado con la representación, incluye además procesos de abstracción, de modo que formulemos el problema adecuadamente, centrándonos en las cuestiones importantes y dejando de lado los detalles.

Los procesos de planificación y ejecución equivalen a la creación de algoritmos definida en el pensamiento computacional, ya que consiste en la elaboración de un plan a partir del establecimiento de objetivos parciales (OECD, 2010). Estos procesos, planificación y creación de algoritmos, incluyen la elección de una estrategia.

Por último, los procesos de monitorización y reflexión, hacen referencia al procesamiento de datos, en especial a los procesos de recopilación y análisis, de cara a realizar la comprobación o análisis de los objetivos parciales y la detección de eventos no esperados (OECD, 2010).

Una forma de representar la secuencia tal y como se plantea en el marco de PISA es como aparece en la figura 1. Si atendemos a los procesos descritos en el pensamiento computacional, pero teniendo en cuenta en todo momento la equivalencia de esta forma de denominar a estos procesos con la denominación de dichos procesos en la literatura de la resolución de problemas, podemos ver la representación del proceso en la figura 2.

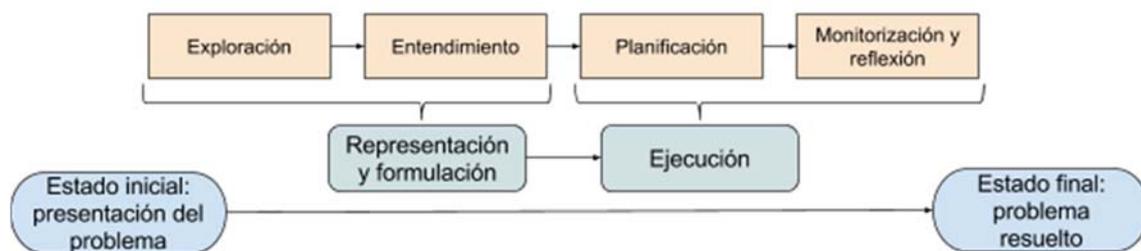


Figura 1. Representación del proceso de resolución de problemas con los términos empleados en PISA 2012 (OECD, 2010). Fuente: elaboración propia

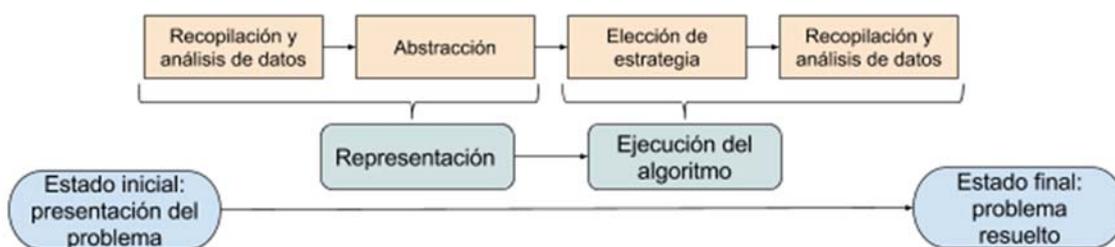


Figura 2. Representación del proceso de resolución de problemas con los términos empleados en el pensamiento computacional (CSTA y ISTE, 2011). Fuente: elaboración propia

Esta representación esquemática es una aproximación que nos ayuda a entender la resolución de problemas y la forma en la que el pensamiento computacional, a través del empleo de una estrategia computacional para la resolución de problemas, mejoraría los procesos de resolución de problemas.

Sin embargo, no es una representación exacta del proceso. La abstracción, como veremos más adelante, se encuentra durante todo el proceso, ya que necesitamos la abstracción para detectar el foco del problema, y precisamos de una continua evaluación de si el foco detectado y los diferentes estados del problema son los adecuados.

Gracias al proceso continuo de recopilación, análisis y representación de datos, podemos evaluar constantemente si estamos siguiendo el camino adecuado para resolver el problema; y si la representación que hemos construido del problema a partir de nuestra abstracción es correcta. Como ya hemos comentado, este proceso se explicará con detalle más adelante.

Como podemos ver, el pensamiento computacional requiere de ciertas características o procesos que no se consideran imprescindibles en el proceso común de resolución de problemas (CSTA y ISTE, 2011).

Entre estos procesos o características que recoge el pensamiento computacional y no vemos en el proceso de resolución de problemas, destaca la *descomposición del*

problema, necesaria para poder afrontar el problema con éxito. Esta descomposición del gran problema en pequeños problemas o *submetas*, permite que algunos de estos pequeños problemas se puedan abordar de forma independiente: algunos se podrían automatizar, otros se podrían resolver o avanzar en ellos de forma paralela y simultánea, etc.

Por tanto, entre las acciones que se llevan a cabo en el paso a paso que definamos para nuestro proceso de resolución, el pensamiento computacional nos proviene de técnicas para la *automatización*, el *paralelismo* y la *simulación*, que facilitan este proceso, y que son posibles gracias a la descomposición del problema, al abordarlo como pequeños cómputos (de ahí, el calificativo de computacional). Estas técnicas o características son propias por tanto del pensamiento computacional.

Por tanto, según esta primera aproximación al concepto, la clave del pensamiento computacional residiría en la descomposición del problema, la automatización, el paralelismo y la simulación; entendiendo estos conceptos como:

- Descomposición del problema: acción de dividir el problema en pequeñas partes o subproblemas independientes, que permitan afrontar cada uno de estas partes por separado, facilitando la resolución.
- Automatización: ejecutar las tareas repetitivas a través de técnicas que permitan la sistematización, generalmente (pero no necesariamente) con medios tecnológicos como ordenadores.
- Paralelismo: planificar las tareas independientes, es decir, los diferentes subproblemas, para que se ejecuten de forma simultánea disminuyendo el tiempo del proceso de resolución.

- Simulación: acción necesaria para comprobar, en entornos supuestamente similares, que las distintas soluciones de los subproblemas se adecúan con las demás soluciones y que, entre todas, logran solucionar nuestro problema meta.

Como vemos por la definición de cada uno de los procesos implicados únicamente en el pensamiento computacional, todos ellos se tornan de gran utilidad como parte de una estrategia computacional, es decir, todos ellos están implicados en la ejecución del algoritmo.

Es decir, según la propuesta de CSTA y ISTE (2011) para los procesos implicados, el pensamiento computacional facilita la resolución del problema gracias a poder abordar el proceso de resolución de forma computacional.

Como ya comentamos a lo largo del capítulo, Wing (2014) defiende que el pensamiento computacional está implicado en la representación y formulación de los problemas, por lo que el PC no sólo estaría implicado en el proceso de resolución como se ha sabido hasta la fecha.

En los siguientes capítulos vamos a profundizar más en cómo este pensamiento podría darse también en la representación del problema.

Capítulo 2. Resolución de problemas complejos

Estado actual de la investigación sobre la resolución de problemas

Desde el origen de la investigación en resolución de problemas, se ha partido de la definición de problema para orientar la búsqueda sobre qué se necesita para resolverlo. Duncker (1945) propuso la siguiente definición: “un problema surge cuando una criatura viviente tiene un objetivo pero no sabe cómo alcanzarlo” (p.1), definición que sigue vigente (Bassok y Novick, 2012, p. 413; Weisberg, 2006, p. 123).

A partir de esta definición, deducimos que lo que para una persona puede ser un problema, puede no serlo para otra o para la misma persona en otro momento porque sí sepa cómo alcanzar el objetivo.

Algunas veces, la persona puede tener un objetivo sin que le suponga un problema, por ejemplo tener hambre e ir a un restaurante, o surgiendo otro problema diferente, siguiendo con el ejemplo, decidir entonces qué restaurante elegir (Bassok y Novick, 2012, p. 413).

Weisberg (2006) alega que, para resolver un problema, la persona debe ser capaz de aplicar una secuencia de operadores que cambian el problema en su estado original hasta su solución, es decir, hasta alcanzar el objetivo. Esta secuencia de operadores es la solución del problema. Si no encuentra un camino para transformar el problema hasta alcanzar el objetivo, no podrá resolver el problema.

Partiendo de esta base, la investigación en el campo de la resolución de problemas ha evolucionado en cuanto a cómo se enfocan los estudios para conocer más sobre este subcampo del pensamiento.

Tradicionalmente, se investigaban la representación del problema y la búsqueda de pasos para resolver el problema como dos temas de investigación diferentes.

Por una parte, se investigaba la representación del problema con problemas cuyos componentes están mal definidos, de forma que los resultados reflejan diferencias en la representación. Un ejemplo de los problemas que utilizaban estos estudios es el de los pájaros y los trenes (Posner, 1973; citado en Bassok y Novick, 2012, p. 415), en el que algunas personas representan el problema centrándose en el pájaro, mientras que otras personas se centran en los trenes.

Por otra parte, la búsqueda de pasos se ha venido investigando con problemas cuyos componentes están bien definidos, por tanto los resultados reflejan cuáles son las mejores estrategias que permiten hallar el mejor camino (Bassok y Novick, 2012). Un ejemplo de este tipo de problemas es el clásico puzle de la torre de Hanoi, que inventó el matemático francés Édouard Lucas en 1883.

Actualmente, el interés sobre la resolución de problemas se da como concepto y como tema de investigación (Funke, 2013). El avance que ha experimentado este campo ha centrado las líneas de investigación en estudios que integran la investigación sobre la representación del problema y la investigación sobre búsqueda. Algunas de estas líneas son trabajo en problemas isomorfos, experiencia de dominio y soluciones por insight. (Bassok y Novick, 2012)

Pero, ¿por qué se ha dado esta integración? Para responder a esta pregunta, debemos hacernos otra: ¿cómo se relaciona la representación del problema con la estrategia seguida para resolverlo?

La representación y la búsqueda de los pasos a seguir para resolver el problema siguen un *proceso recursivo* hasta que se soluciona el problema (Bassok y Novick, 2012).

Este proceso ocurre cuando, al ejecutar algún paso para resolver el problema, la persona que lo resuelve se da cuenta de que no ha tenido en cuenta un componente del problema en su representación, por lo que tiene que actualizar la representación del problema incluyendo dicho componente y posiblemente esta acción modifique el proceso a seguir para alcanzar la meta.

Este proceso se desarrolla, como hemos dicho, de forma recursiva. Esto significa que continuamente puede ocurrir que incluyamos nuevos componentes y tengamos que reformular el problema. Un ejemplo de esta relación se muestra en la investigación que publicó Duncker por primera vez en 1935, en la que analizó las soluciones de estudiantes de medicina sobre el problema de la radiación (Radiation Problem), soluciones que demostraban esta relación entre representación y búsqueda (Holyoak, 2012).

A continuación, vamos a desglosar tanto la representación del problema como el proceso para la resolución del problema, de forma que nos permita recoger todo lo que se ha investigado sobre los factores que influyen en la resolución de problemas.

De este modo, nos ayudará a determinar en qué momento aparecen los procesos cognitivos propuestos en el pensamiento computacional que están directamente relacionados con la resolución de problemas: abstracción, recopilación de datos, análisis de datos y representación de datos, y, por último, uso de algoritmos.

Representación del problema

¿Cómo se forma la representación de un problema?

Bassok y Novick (2012) definen la *representación de un problema* como un modelo construido por la persona que soluciona el problema y que recoge todo su entendimiento de los componentes del problema: el estado inicial, el estado objetivo y el

conjunto de posibles operaciones que podría aplicar para conseguir llegar del estado inicial al estado objetivo (p. 414).

El concepto de los modelos mentales tiene antecedentes desde el siglo XIX. Sin embargo, no es hasta el siglo XX cuando se propone la teoría de los modelos mentales (Johnson-Laird, 1980). El autor propone la teoría para explicar los procesos superiores de la cognición, concretamente la comprensión y la inferencia.

La teoría de los modelos mentales explica principalmente la forma de representar el mundo, pero también incluye una explicación sobre los procedimientos que permiten construir y manipular los modelos representados. Entre estos procedimientos se incluyen una mente computacional, procedimientos efectivos, revisión recursiva y los propios modelos mentales (Rodríguez Palmero, Marrero Acosta y Moreira, 2001).

Esta teoría se basa en explicaciones entorno al razonamiento deductivo, razonamiento probabilístico, y razonamiento inductivo y modularización, entendida como la capacidad de descomponer en pequeños módulos o acciones mínimas computables. Además, incluye un proceso de detección de inconsistencias seguido de otro proceso que permite establecer las causas y crear explicaciones, salvando las inconsistencias detectadas (Johnson-Laird, 2012).

Para explicar cómo se forma el modelo creado sobre el problema, es decir, la representación del problema, Reitman (1965) afirma que hay determinados componentes de un problema que influyen en la interpretación, y que difieren en el grado en el que estén bien definidos, un posible caso sería si la situación tiene los detalles bien definidos y no es necesario un gran trabajo de interpretación, el otro caso posible sería si hay componentes mal definidos y entra en juego la capacidad de definirlos de la persona que resuelve el problema (Bassok y Novick, 2012, p. 414).

De esta afirmación podemos extraer dos factores clave que influirán en la representación del problema: el contexto y el conocimiento previo de la persona que resuelve el problema.

Factores que influyen en la representación del problema

Existen diversos factores que influyen en cómo representamos el problema. Estos factores son externos e internos a la persona que resuelve el problema. Como principal factor externo nos encontramos con el contexto del problema, el cual influye de varias formas que veremos a continuación. Como principal factor interno nos encontramos el conocimiento previo, ya que, como veremos próximamente, la información que tenga en su memoria será clave para representar el nuevo problema.

Contexto

El efecto del contexto puede afectar a la representación de tres formas diferentes (Novick y Bassok, 2005): por la forma en la percepción, las inferencias que resultan de la familiaridad con dicho objeto o por el contenido de la historia del problema.

Un ejemplo clásico que ilustra de forma clara el primer efecto que afecta a la representación, la forma en la percepción, es el problema de los nueve puntos (Maier's, 1930), en el que debemos unir los nueve puntos a través de tan sólo 4 líneas. En este problema, las personas crean un límite imaginario con los puntos externos, creando un cuadrado de 3x3 puntos del que creen que no pueden traspasar el límite creado por dicho cuadrado (Novick y Bassok, 2005, 2012; van Steenburgh, Fleck, Beeman, y Kounios, 2012).

Este fenómeno impide que los sujetos que intentan resolver el problema de los nueve puntos sean capaces de hacerlo, hasta que no son conscientes de que dicho límite

lo han impuesto ellos mismos. Es entonces cuando sucede una reestructuración del problema y la solución se vuelve fácil.

Por otra parte, existen inferencias que se basan en la funcionalidad de un objeto y se deben a la función que asignamos a un objeto familiar. Este tipo de inferencias han sido estudiadas como el fenómeno de fijación funcional (Dunker, 1945).

Dunker (1945) investigó cómo las personas definían algunos de los componentes, demostrando el fenómeno de fijación funcional que postula que si un objeto es usado normalmente con un propósito cerrado y preciso, es más difícil ver que ese objeto puede tener propiedad que le hagan capaz de ser usado para un propósito diferente (Bassok y Novick, 2012, p. 417).

Algunos ejemplos de este fenómeno son el problema de las velas (Dunker, 1945) y la investigación sobre el problema de las dos cuerdas, llevada a cabo por Birch y Rabinowitz (1951).

En el problema de las velas (Dunker, 1945), a los sujetos se les daba una serie de objetos para resolver un problema, entre ellos una caja. Si la caja contenía el resto de objetos, los sujetos no la tenían en cuenta como un componente más para la solución del problema, mientras que si la caja se les daba vacío desde el principio se utilizaba como un componente más para resolver el problema.

En la investigación realizada por Birch y Rabinowitz (1951) se demuestra que el fenómeno de fijación funcional puede darse por un reciente uso del objeto. En la investigación, se les proponía una serie de objetos para realizar el problema clásico de Maier's (1931, citado en Bassok y Novick, 2012, p. 417) de las dos cuerdas. Luego se les proponía otra tarea que podía resolverse con los mismos objetos propuestos para el

problema de las cuerdas. Los resultados demostraron que los sujetos evitaban usar el mismo objeto para dos funciones no relacionadas (Bassok y Novick, 2012, p. 417).

Con estos dos fenómenos se demuestra que los componentes que aparecen en un problema tienden a evocar inferencias de otros componentes no explícitos, que la gente incorpora a su representación del problema, y que en muchas ocasiones estos componentes inferidos no pertenecen realmente al problema al que se enfrentan.

El tercer y último motivo que puede afectar a la representación del problema es el contenido de la historia del propio problema. Para estudiar este fenómeno fue necesario la evolución de la investigación sobre resolución de problemas, en la que se toma la representación del problema y la búsqueda de heurístico como un proceso recursivo.

Una de las líneas de investigación que se ha centrado en estudiar este fenómeno es el estudio de problemas isomorfos, que consiste en usar problemas bien definidos, tanto los clásicos como variantes del problema, en los que varía el contenido. De esta forma, la estructura del problema y el espacio problema es el mismo, y las diferencias dependerán de las inferencias de perceptivas y conceptuales (Bassok y Novick, 2012, p. 420).

Un ejemplo de problema isomorfo que se ha utilizado para demostrar este fenómeno es el de la Torre de Hanoi y la creación de dos variantes cuyo contenido era relacionado con Monstruos y Globos (Hayes y Simon 1977, citado en Bassok y Novick, 2012, p. 420). En la primera variante, las reglas eran igual que en el problema clásico de la Torre de Hanoi, y así se le explicaba a los sujetos. En la segunda de las variantes se les decía como regla que los monstruos podían contraerse y expandirse acuerdo a unas reglas, reglas que equivalían a las de la primera variante.

Los sujetos realizan la búsqueda entre una variante y otra del problema en dos caminos cualitativamente distintos, es decir, la representación de su espacio problema y

el movimiento de los sujetos por éste era diferente en cada variante. Los investigadores concluyeron que las diferencias se deben a que la segunda variante era más fácil de imaginar y, por tanto, el problema era más fácil de resolver.

Otro ejemplo con resultados relacionados con los que acabamos de exponer también se basa en una variante de la torre de Hanoi, esta vez los sujetos debían hacerlo con una pirámide invertida, imaginando que eran acróbatas (Kotovsky, Hayes y Simon, 1985; citado en Bassok y Novick, 2012, p.421). De forma similar, los resultados muestran una mejor representación facilitando la solución del problema.

Otro ejemplo más reciente sobre cómo afecta el contenido del problema es el estudio realizado por Chu, Li, Su y Pizlo (2010) que muestra cómo si el contenido resulta más familiar los participantes tienen mayor frecuencia de éxito en la resolución del problema y tardan menos tiempo en resolverlo.

Gracias a estos tres tipos de fenómenos, que recordamos son: la forma en la percepción, las inferencias basadas en el objeto y el contenido del problema, se ha podido demostrar la influencia del contexto en la resolución de problemas.

Conocimiento previo

El estudio sobre la influencia del conocimiento previo en la representación del problema se ha venido realizando, principalmente, comparando los resultados entre expertos y novatos en experiencias de dominio (Chase y Simon, 1973; Gobet y Simon, 1996; Hinsley, Hayes, y Simon, 1977; Paige y Simon, 1966; Larkin, Mc Dermott, Simon, y Simon, 1980; Simon y Simon, 1978; ver en Bassok y Novick, 2012).

La principal diferencia entre expertos y novatos reside en los patrones de reconocimiento que emplean unos y otros al enfrentarse a un problema. Mientras que los novatos centran su atención en aspectos no relevantes y advierten los componentes del

problema de forma aislada, los expertos se centran en aspectos relevantes y además crean configuraciones familiares con los componentes que observan del problema (Bassok y Novick, 2012).

Estas configuraciones familiares, han sido denominadas *chunks* (trozo, en español) de información, y se podrían definir como categorías, es decir, grupos de información que se relacionan entorno a un significado (Rips, Smith y Medin, 2012).

Debemos ser cautelosos ante la aparición de los conceptos *chunks* y *chunking*, ya que han sido utilizados en diversas áreas relacionadas o no con la psicología, y podría no significar lo que esperamos. Entre estas áreas se encuentran la memoria (psicología), en la que nos encontramos situados nosotros, la acción motor (percepción), y varios significados relacionados con la arquitectura cognitiva, y en cuanto a otros significados del término, éstos se relacionan con disciplinas como las ciencias de la computación, la lingüística o la educación (Gobet, Lloyd-Kelly y Lane, 2016).

Gracias a esas asociaciones entre componentes relevantes del problema, los expertos son capaces de proponer rápidamente un plan de solución (Anderson, 1982; citado en Bassok y Novick, 2012). Si, además, la organización del conocimiento es óptima, el acceso a la información en la memoria a largo plazo se facilita ya que la relación entre ésta y la memoria de trabajo es más fluida (Ericsson y Kintsch, 1995).

Mientras, los novatos tendrían que emplear mucho más tiempo en computar esos componentes, es decir, en compilarlos de forma que puedan proponer un procedimiento similar para solucionar el problema (ver Chase y Simon, 1973; Novick y Sherman, 2003, 2008; citados en Bassok y Novick, 2012).

Como exponen Bassok y Novick (2012), esta categorización se basa en extraer la estructura del problema que es relevante para la solución del mismo, por ejemplo, a través

de establecer relaciones causales (Cheng y Buehner, 2012). Por tanto, permite la búsqueda en la memoria de problemas estructuralmente análogos, tanto del mismo dominio de conocimiento como de otros dominios, en el último caso siempre que se trate de dominios en los que no se requiere un alto conocimiento específico.

Una vez que se ha encontrado un problema análogo en este sentido, la solución del problema ya resuelto se puede aplicar o adaptar al nuevo problema. En este punto tiene la base la teoría de los modelos mentales de Johnson-Laird (1980).

Aunque la categorización puede no ser correcta, los expertos tienen una mayor capacidad para detectar pistas engañosas (ver Hegarty, Mayer y Green, 1992; Lewis y Mayer, 1987; Martin y Bassok, 2005; Novick 1988, 1995; Novick y Holyoak, 1991; citados en Bassok y Novick, 2012), en cuyo caso serán más rápidos para detectar la no correspondencia y desechar esa solución.

Una vez introducida la importancia del conocimiento previo en la representación del problema, pasemos a ver cómo influye específicamente la memoria y la comprensión. Mientras que la memoria se relaciona con los mecanismos específicos de dominio que permiten resolver el problema, la comprensión tiene relación con los mecanismos de dominio general que ponen en marcha procesos de abstracción (Chiappe y MacDonald, 2005).

Memoria

La memoria es un sistema mental cuya función principal es almacenar información. Este sistema está compuesto a su vez de dos subsistemas, memoria de trabajo, originalmente denominada memoria a corto plazo, y memoria a largo plazo (Baddeley y Hitch, 1974; Baddeley, 1992).

Existen numerosas evidencias de que una de las funciones principales de la memoria de trabajo es mantener representaciones de la memoria a largo plazo activas y accesibles con un alto rendimiento (Hambrick y Engle, 2003). Tal y como señala Morrison (2005) la memoria de trabajo es el “constructo cognitivo responsable del mantenimiento y manipulación de información y es necesario para muchos de los tipos de pensamiento complejo como la resolución de problemas” (p. 457).

Como ya hemos comentado anteriormente, las investigaciones sugieren que existe una mejor comunicación entre ambas, más fluida y eficaz, cuanto mayor sea la experiencia de dominio y la organización del conocimiento que dispone una persona (Ericsson y Kintsch, 1995). Bajo esta suposición, cuando se produce un estancamiento en la resolución del problema en expertos, la memoria de trabajo proporciona información desde diversas y más variadas aproximaciones al problema, no limitando la posibilidad de resolución de éste.

Además, la memoria de trabajo juega un rol central en el procesamiento de información (Baddeley y Hitch, 1974), concretamente ejerce un papel fundamental en la recopilación y el análisis de los datos procedentes del exterior, ya que la capacidad de la memoria de trabajo se puede definir como la capacidad general de procesamiento de la información que corresponde al control que ejerce nuestra atención (Wiley y Jarosz, 2012). Por tanto, la correcta recopilación y análisis de la información van a permitir hacer una correcta representación del problema.

Estos dos procesos cognitivos, recopilación de información y análisis de información, se definen como características propias del pensamiento computacional (CSTA y ISTE, 2011). La recopilación tendría lugar cuando recogemos nuevos datos del entorno, mientras que el análisis tendría lugar cuando intentamos establecer relaciones entre los datos obtenidos y datos similares almacenados en la memoria a largo plazo.

Por otra parte, la memoria tiene implicaciones en la representación del problema de tres formas diferentes (Novick y Bassok, 2005). En primer lugar, encontramos la experiencia con problemas estructuralmente similares o análogos. En segundo lugar, se sitúan los esquemas generales para tipos de procedimientos para llegar a la solución y tipos de herramientas comunes de representación. En tercer y último lugar, encontramos la experiencia de dominio.

Hasta ahora hemos hablado del primer y tercer tipo de implicaciones de la memoria en la representación del problema. Si los problemas tienen una estructura similar o un contenido similar, es decir, contenido específico de dominio, nuestra memoria juzgará esa similaridad para aplicar procedimientos de solución del problema similares a los que aplicó en ocasiones anteriores para resolver los problemas análogos.

En este punto debemos destacar la importancia del segundo tipo, de los esquemas generales en la memoria necesarios para representar el problema. Se trata de esquemas abstractos que guardamos en nuestra memoria y que se utilizan para determinados tipos de problema. Estos esquemas son utilizados de forma que excluyen la información idiosincrática, realizando un proceso de abstracción que permite fijarnos únicamente en el tipo de información particular que es común a ambos problemas.

Entre estos esquemas abstractos, encontramos en la literatura tres tipos de diagramas espaciales: matrices, redes semánticas y jerarquías, siendo importantes herramientas para entender y resolver problemas de una gran variedad de dominios (Novick y Bassok, 2005).

Como afirma Tversky (2005), los diagramas facilitan la resolución de problemas, ya que ayudan a crear inferencias espaciales que permiten establecer relaciones lógicas entre conceptos diferentes, en concreto facilita las relaciones estructurales y funcionales. Existen estudios que muestran una relación entre el razonamiento espacial y el

pensamiento computacional, este último medido a través de las calificaciones en programación (Ambrosio, Almeida, Macedo y Franco, 2014).

Un diagrama bien construido permite, por tanto, extraer inferencias sobre las cualidades y las relaciones entre conceptos, a partir del estudio de propiedades básicas del diagrama como pueden ser la distancia, dirección o tamaño (Tversky, 2005).

Por tanto, como hemos visto en este apartado, la memoria juega un papel fundamental en la recopilación y el análisis de nueva información. En un primer lugar, sólo recopilamos la información que nos parece relevante según nuestra atención selectiva debido a las limitaciones de nuestra memoria de trabajo.

En segundo lugar, la memoria tiene importantes implicaciones para la representación del problema, ya que la información que nuestra atención juzga como relevante es la información que tiene similitud con la información recogida para otros problemas que consideramos similares, ya sea por la estructura o por el contenido.

Además, gracias a los esquemas abstractos que utiliza nuestra memoria, podemos resolver una gran diversidad de problemas utilizando estos esquemas o procedimientos. Estos recursos memorísticos nos ayudan a analizar la información de cara a hacer una mejor abstracción y poder comprender y contextualizar la nueva información, siendo capaces de establecer nuevas relaciones estructurales y funcionales.

Comprensión

En este apartado vamos a tratar diversos temas relacionados con la comprensión del problema y su correcta representación, que influirá inevitablemente en el éxito o el fracaso de conseguir solucionar el problema.

En primer lugar vamos a hablar de la importancia de la abstracción y la necesidad de su adecuación a la correcta representación del problema en términos más abstractos y

generales. En segundo lugar, vamos a relacionar una adecuada comprensión del problema con la importancia de la representación adecuada del mismo. En tercer y último lugar, revisaremos las investigaciones sobre el concepto de insight, considerado tradicionalmente como un elemento imprescindible para la resolución de problemas.

La abstracción es un proceso cognitivo en el que se basa, en gran medida, la comprensión del problema. Una correcta comprensión conducirá a una planificación adecuada, es decir, planificaremos a partir de la abstracción (Polson y Jeffries, 2014, p.423). El proceso de abstracción es crucial en la recopilación y análisis de información. Ya que el análisis que realizamos de los datos obtenidos, debe facilitar la comparación con la información ya almacenada en la memoria, de cara a detectar similitudes y casos análogos, y hacer una adecuada planificación. Por este motivo, la abstracción es necesaria para facilitar la detección de similitudes y planificar.

Este proceso incluye descartar la información irrelevante hasta quedarnos únicamente con la información generalizable, es decir, la información que es relevante para nosotros porque nos permite generalizar el problema y hacerlo más similar a problemas que hemos resuelto en anteriores ocasiones. De este modo, la abstracción es la que nos permite hacer un correcto análisis de la información.

Como ya señalaba Weirthelmer (1959), existen diferencias entre los estudiantes que comprenden y los que no comprenden el problema. A partir de un problema en el que los estudiantes tenían que hallar el área de un paralelogramo, este autor se dio cuenta de que quienes comprendían la fórmula exhibían un pensamiento productivo, ya que eran capaces de entender la lógica que subyace a la fórmula, es decir, de abstraer la información importante del problema (convertir el paralelogramo en un rectángulo).

Sin embargo, quienes no eran capaces de comprenderla, aplicaban la fórmula a nuevos paralelogramos a partir de la memorización de la solución hallada en el problema

original. Siendo capaces únicamente de resolver los problemas similares al original, exhibiendo un pensamiento reproductivo (Bassok y Novick, 2012).

Para el proceso de abstracción nos ayudamos de esquemas abstractos, o formas de representación, que disponemos en la memoria, como ya comentamos anteriormente. Fundamentalmente diagramas espaciales tales como matrices, redes semánticas, jerarquías o mapas (Tversky, 2005).

Para comprender la información, necesitamos representar la información, de forma que podamos conectarla con nuestro conocimiento previo. Siguiendo el modelo de comprensión que recoge Sánchez Miguel (2008), la comprensión profunda implica la construcción de representaciones mentales incluidas dentro de representaciones más generales de conocimiento. Esta red de conocimientos donde se integra la nueva información es necesaria para que se produzca una representación adecuada del problema.

El proceso de abstracción que hemos tenido que llevar a cabo para representar la información, hará que nos resulte mucho más sencillo representarla, ya que nos centraremos únicamente en representar la información relevante, facilitando este proceso. Además, la información que elegimos para incluir en la representación, tiene una gran influencia en cómo se interpreta y procesa la información, ya que activará conocimientos concretos (Hegarty y Stull, 2012).

La intención de representar la información es darle significado (Markman, 2012), y esto es posible gracias a un conjunto de relaciones que establecemos para corresponder el mundo representado con el mundo a representar. En ese conjunto de relaciones representativas se encuentran las que relacionan la información con lo ya conocido.

De esta forma, se consigue dotar la nueva información de significado incluyendo la nueva información a esquemas, o representaciones de conocimiento, propios de la persona.

Para lograr una correcta representación de la información del problema, tenemos a nuestra disposición diferentes formas complementarias de representación que permiten, a través de determinados procesos, establecer las relaciones entre el mundo a representar y el mundo que representamos que hemos determinado como necesarias para representar el conocimiento (Markman, 2012).

Estas formas de representación son diferentes para que podamos disponer de la que más nos convenga para cada tipo de información. Como afirma Markman (2012), probablemente usemos una variedad de formas de representación para hacer frente a cada problema.

Además, cada forma de representación parece requerir un tipo específico de tareas cognitivas por lo que la complementariedad es la clave para la existencia de diversas formas de representación, en lugar de disponer de una única forma de representación. Se viene reclamando un pluralismo en la representación (Dove, 2009).

Markman y Dietrich (2000) discuten los caminos generales para diferenciar unos tipos de representaciones de otras, siendo principalmente tres los aspectos discriminatorios.

El primero de ellos hace referencia a si la representación utiliza símbolos discretos o usa un espacio continuo. El segundo aspecto está relacionado con los elementos de la representación, dependiendo de si estos elementos son independientes o incluyen conexiones específicas y por tanto están relacionados. El tercer aspecto que permite

discriminar entre los tipos de representación es si la representación es analógica o simbólica respecto al mundo a representar (Markman y Dietrich, 2000).

Estos aspectos nos hacen diferenciar entre representaciones espaciales, representaciones de características y representaciones estructuradas, perteneciendo a estas últimas las redes semánticas y las representaciones relacionales estructuradas (Markman, 2012).

Los procesos de abstracción y representación de información influyen indudablemente en la solución del problema. Para llegar a ésta, se requieren procesos analíticos que permitan crear una solución satisfactoria a partir de la información representada.

A partir de este supuesto, el pensamiento computacional debería estar implicado también en la representación del problema, ya que según cómo se represente éste, resultará más o menos fácil establecer el procedimiento para la resolución del mismo.

Por otra parte, también se ha venido defendiendo la existencia de un proceso oculto, inconsciente, que permite dar con la solución del problema a través de un momento clave, podríamos decir un momento de lucidez, un momento de comprensión súbita del problema. Este proceso fue propuesto por la escuela de la Gestalt y ha sido tradicionalmente conocido como *Insight*.

El *insight* ha sido ampliamente estudiado desde su propuesta y la investigación actual también ha querido desmitificar el concepto de insight en la resolución de problemas (Mayer, 1995; Bowden, Jung-Beeman, Fleck y Kounios, 2005; van Steenburg et al., 2012).

Desde los años 80, diversas investigaciones han mostrado que las soluciones por insight no surgen necesariamente de repente ni como un estallido tras la reestructuración

(Weisberg y Alba, 1981); e incluso cuando lo hacen, el proceso de solución subyacente (en este caso fuera de la consciencia) podría reflejar un progreso incremental hacia el objetivo (Bassok y Novick, 2012, p. 423).

Van Steenburg, Fleck, Beeman y Kounios (2012) revisan la investigación sobre la resolución de problemas a través del insight frente a la resolución de problemas con procesos analíticos y concluyen que ambos procesos, los relativos al insight y los procesos analíticos, ponen en marcha las mismas habilidades centrales, como la memoria de trabajo, la inteligencia fluida, la habilidad general de resolución de problemas y vocabulario.

Por otra parte, estos autores (van Steenburg et al, 2012) apuntan a que existe suficiente evidencia acumulada para confirmar la demostración de que el hemisferio cerebral derecho contribuye a la solución de problemas por insight que no ha podido ser demostrada en procesos analíticos, por lo que la comunidad científica acepta el insight como un proceso especial y diferencial al proceso analítico para resolver problemas (Weisberg, 2015; Chuderski, 2014).

Hasta aquí hemos revisado los factores que influyen en la representación del problema, contexto y conocimiento previo. En el primer caso resumimos las tres formas en las que el contexto puede influir en la representación del problema. En el segundo caso, el conocimiento previo, hemos visto cómo se investiga actualmente su influencia a través de los estudios de expertos y novatos, y hemos conocido cómo afecta la memoria y la comprensión del problema en su representación. A partir de ahora nos centraremos en el proceso de resolución de problemas.

Proceso para solucionar el problema

En cuanto al proceso empleado para solucionar el problema, existen dos modos en los que se da la búsqueda de pasos, el empleo de algoritmos y el uso de heurísticos. El uso de algoritmos, o método de búsqueda a través del espacio problema, recae habitualmente en los novatos, en gente sin experiencia de dominio. Por su parte, el empleo y selección de heurísticos es más común en expertos (Bassok y Novick, 2012).

De esta forma, podemos exponer que el factor fundamental que afecta a cómo realizamos la búsqueda de pasos para la solución del problema es la experiencia de dominio. Un argumento a favor de que el uso de heurísticos es más común en expertos lo encontramos en el estudio realizado por Kotovsky y Simon (1990). Estos autores encontraron que la transferencia de habilidades de un problema a otro, era fundamental para resolver un problema, concretamente utilizaron el rompecabezas chino de los anillos.

Según sus características entorno al tamaño del espacio problema y la cantidad de conocimiento requerida, era un problema isomorfo a otros problemas presentados en el estudio. Sin embargo, este problema fue mucho más complicado por la escasa transferencia de los operadores de movimiento adecuados (Kolovsky y Simon, 1990).

Algoritmo como búsqueda a través del espacio problema

La búsqueda de pasos para resolver el problema se ha venido investigando desde hace tiempo. Newell y Simon's (1972) definieron *espacio problema* como es espacio que hay entre el estado inicial del mismo, el conocimiento que tiene la persona que resuelve el problema en el momento en el que le presentan el problema, y su resolución.

El espacio problema contiene un conjunto de estados de conocimiento, un conjunto de operadores que permiten moverse entre los diferentes estados de conocimiento e información local sobre el camino que se está eligiendo para avanzar entre

los estados, es decir, si se está más cerca de solución el problema o no (Newell y Simon, 1972).

El conjunto de estados de conocimiento contiene dos estados bien conocidos: el estado inicial del problema y el estado objetivo del problema, pero además alberga varios posibles *subestados* del mismo que permiten avanzar en la solución del problema (Newell y Simon's, 1972).

Un algoritmo es un método que garantiza la solución de un problema (Novick y Bassok, 2005). Un método serial para hacer progresos incrementales hacia el objetivo, aplicando operadores que nos permiten avanzar desde el estado inicial hacia el estado objetivo del problema.

Recoge todos los pasos necesarios a muy bajo nivel, lo que permite evitar la mayoría de elementos adversos. Este nivel de detalle de cada uno de los pasos que contiene el algoritmo pretende no dejar nada a la interpretación de la persona que resuelve el problema, ya que esta interpretación puede inferir elementos erróneos que le desvíen de la solución adecuada.

Idealmente, sería adecuado el uso de algoritmos para resolver cualquier tipo de problema, dada la planificación del proceso a bajo nivel, que como hemos dicho permitiría evitar elementos adversos, y tener más probabilidades de alcanzar con éxito la solución del problema. Sin embargo, el uso de algoritmos requiere demasiado tiempo para la planificación y ejecución de todos los pasos, dada la complejidad que reside en los problemas habituales. Por tanto, en pocas ocasiones se puede emplear el uso de algoritmos para la resolución de problemas.

Empleo y selección de heurístico

Un heurístico es una estrategia, un atajo que toma una persona cuando está bastante segura de que el atajo le va a permitir resolver el problema (Novick y Bassok, 2005).

Newell y Simon's (1972) descubrieron el empleo de heurísticos cuando realizaron sus investigaciones para comprobar el proceso de resolución que empleaban las personas, intentando descubrir regularidades en el paso a paso desde el estado inicial hasta el estado objetivo del problema.

Cuando se investiga sobre un proceso cognitivo, se emplean protocolos de pensamiento en voz alta. En las personas que tienen familiaridad con el problema, los pasos verbalizados no son todos los necesarios para alguien que no tiene familiaridad con el problema (Novick y Bassok, 2005). Por tanto, el empleo de heurístico se da cuanto mayor es la experiencia en un campo, en un dominio específico.

Sin embargo, Newell y Simon's se basan en unas dimensiones para la investigación sobre el empleo de heurísticos que no son particulares en la resolución de problemas, sino que son comunes a otros procesos cognitivos superiores. Para Ohlsson (2012) esto podría suponer un problema porque puede poner en peligro la investigación del empleo de heurísticos en la RP.

Para superar esta limitación de la investigación en RP, Ohlsson (2012) propone unas nuevas dimensiones de lo que él considera una posible nueva teoría de resolución de problemas, cuyas cinco dimensiones son la percepción del problema, la definición del problema o de los objetivos del problema, la evaluación o juicio de los resultados, la toma de decisiones y selección de la acción, y la reparación o subsanación de acciones.

Esta posible teoría es por ahora algo hipotético a futuro, por lo que su inclusión en esta investigación tiene el único objetivo de estar al día de las limitaciones y las propuestas de superación de éstas que encontramos a lo largo de la literatura actual sobre RP.

Por tanto, aunque se destaquen ciertas limitaciones en las investigaciones de Newell y Simon's, éstas han proporcionado una explicación suficiente para comprobar la existencia de unas estrategias de atajo, llamados heurísticos, útiles para la solución de problemas por parte de expertos.

En cuanto a los tipos de heurísticos que se han estudiado en el ámbito de la resolución de problemas, Postigo (2004) recoge las siguientes:

- Búsqueda aleatoria. Consiste en buscar de manera desordenada la solución a un problema, es decir, se dan respuestas al azar esperando encontrar por casualidad la solución del problema. No requiere seguir ningún tipo de planificación y ver si damos con la combinación correcta.
- Ensayo y error. La búsqueda de la solución es ordenada y planificada. Se seleccionan diferentes caminos para encontrar la solución y se van comprobando los resultados de cada uno hasta solucionar el problema. Los resultados fallidos se registran para no repetirlos.
- Búsqueda exhaustiva o sistemática. Se crea un diagrama o <árbol estado-acción> a partir del estado inicial. Consiste en plantear los posibles resultados a partir de ese estado. A continuación, se plantean los posibles resultados de cada uno de los resultados anotados. Y así sucesivamente.
- Subir la montaña o subir la cuesta. Esta estrategia es una modificación de la anterior (búsqueda exhaustiva). La diferencia está principalmente en que sólo desarrollamos

el resultado que consideramos más eficaz y/o el que más nos acerca a la meta, es decir, con el que menos pasos necesitaremos.

- **Análisis de medios-fines.** Consiste en comparar el estado inicial y el estado final o solución del problema, analizar la diferencia entre ellos y encontrar algún tipo de acción que se aproxime y reduzca la distancia entre ambos. Para ello, se divide el problema en submetas que nos aproximan a la solución y se seleccionan las acciones más adecuadas para cada estado del problema (estados entre submeta y submeta).
- **División del problema en subproblemas.** Consiste en fraccionar el problema en varias partes y resolver cada pequeña parte de forma independiente. A diferencia de la estrategia de análisis de medios-fines, esta estrategia no implica que las partes sean fases del proceso del problema. Por tanto, hay que tener cuidado al utilizar esta estrategia, ya que la división del problema puede no ser acertada (por ejemplo, por una representación errónea del mismo) lo que no garantiza la resolución del problema.
- **Búsqueda hacia atrás.** Esta estrategia es similar a la de análisis de medios-fines. La diferencia es que, en este caso, el árbol de estado-acción se empieza desde el estado final del problema y se va explorando hacia el estado inicial. Para aplicar esta estrategia, es necesario que la meta del problema esté suficientemente especificada y que sea una solución única, es decir, que el problema no se pueda solucionar de otra forma.
- **Analogías.** Consiste en utilizar la solución de otro problema diferente pero cuya solución sirve como modelo para solucionar el problema existente. Aunque parezca que esta estrategia es bastante común, los estudios dicen que su uso es poco frecuente porque no siempre es fácil detectar el parecido entre los problemas.

Hasta aquí se ha repasado la investigación en resolución de problemas, atendiendo a su enfoque tradicional, que recordemos se basaba en diferenciar la representación del

problema y el proceso de resolución del mismo, pero incorporando toda la información actual que está directamente relacionada con el tema de conocimiento de esta investigación. A continuación nos centramos en la resolución de problemas complejos, en los que supuestamente influye el pensamiento computacional.

Resolución de problemas complejos (RPC)

Para poder investigar sobre la resolución de problemas complejos, necesitamos partir de una definición consensuada que nos permita abordarlos de una forma más o menos sistemática, similar por todos los investigadores interesados en la resolución de problemas complejos.

A pesar de los intentos por diseñar problemas complejos reales, es necesario un acuerdo previo sobre qué es lo que constituye un problema y qué es lo que constituye la resolución de un problema complejo. Ya que, como apuntan Fischer, Greiff y Funke (2012), la mayoría de las investigaciones sobre RPC se han basado en un enfoque empírico de minería de datos, sin una definición clara y consensuada por la comunidad científica.

Para esta ocasión, vamos a partir de la definición de resolución de problemas complejos que dieron en su momento Funke y Frensch (1995):

“La resolución de un problema complejo ocurre para superar las barreras entre un estado dado o inicial y un estado objetivo deseado por medio de actividades comportamentales y/o cognitivas que incluyen múltiples pasos. El estado inicial, el estado objetivo y las barreras entre el estado inicial y el estado objetivo son complejas, cambian dinámicamente durante la resolución del problema, y no son transparentes. Las propiedades exactas del estado inicial, el estado objetivo y las barreras no son conocidos por la persona que resuelve el problema en el momento de partida. La resolución de un

problema complejo implica la interacción eficiente entre la persona y los requerimientos situacionales de la tarea, e implica unas habilidades y conocimiento cognitivo, emocional, personal y social.” (Funke y Frensch, 1995, p.18)

Estos autores proponen además un marco teórico para la resolución de problemas complejos (Funke y Frensch, 1995). Este marco se basa en asumir que (a) el objetivo teórico de la investigación en RPC es entender la relación entre los factores cognitivos, motivacionales, personales y sociales que se dan en las tareas complejas, y (b) la relación entre los componentes puede ser entendida mejor bajo el modelo de procesamiento de la información.

En este marco teórico se presentan los principales fenómenos que han sido demostrados de forma empírica. Estos fenómenos o factores que interfieren en la RPC son internos y externos, siendo todos ellos aspectos estáticos en la interacción. Con aspectos estáticos se refieren a que la información está disponible desde el principio del problema; aunque estos aspectos en sí sean dinámicos para la persona que resuelve el problema. Funke y Frensch crearon una representación de estos aspectos estáticos de la RPC, que se puede consultar en la figura 3.

Los factores internos son la experiencia; las variables cognitivas, como el conocimiento previo, las estrategias de monitorización y evaluación o el estilo cognitivo; y las variables no cognitivas, como la perseverancia o la motivación.

Los factores externos son la estructura del problema, que incluye aspectos como la complejidad o la transparencia de la tarea; el contexto del problema, es decir, si es familiar y facilita el entendimiento; y los factores ambientales, como si la tarea proporciona feedback, las expectativas, la cooperación, etc.

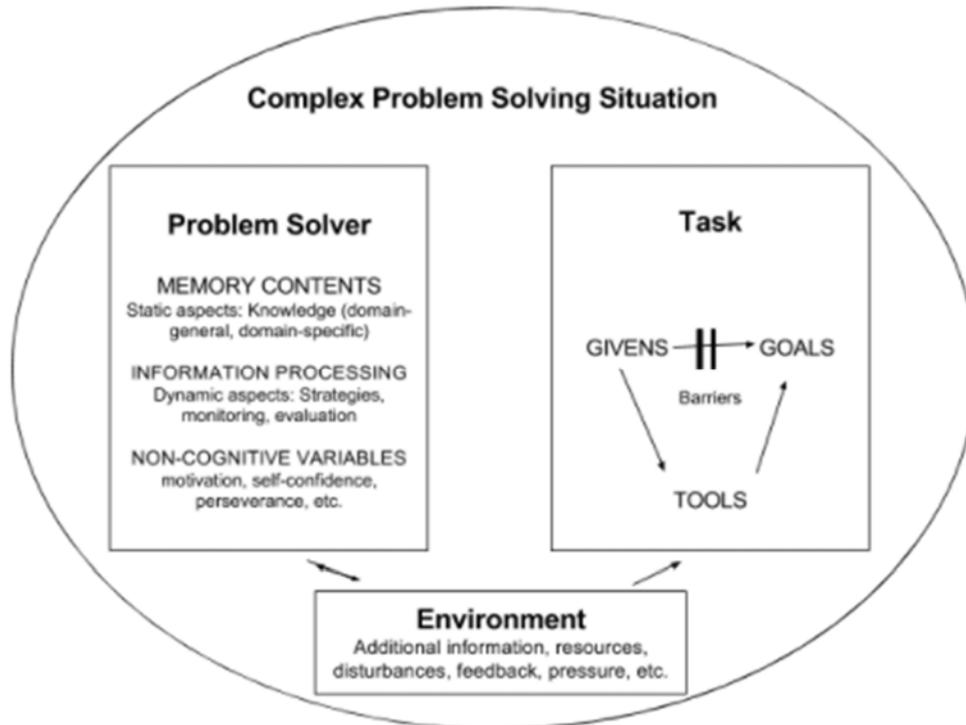


Figura 3. Aspectos estáticos de la interacción producida en una situación de resolución de un problema complejo.
Fuente: Funke y Frensch (1995)

A partir de este marco teórico resulta más fácil el estudio de las características de la resolución de problemas complejos.

Características de la RPC

A partir de la definición propuesta por Funke y Frensch (1995) y de su marco teórico sobre los aspectos que se dan en una situación de RPC, observamos cómo los problemas complejos dependen en gran medida de la interacción continua entre la persona y el estado en el que se encuentre el problema en cada momento, interacción que permita incorporar fácilmente nuevos elementos al problema y permita reformularlo cuando sea necesario.

Desde la perspectiva europea de la RPC, se han definido unas características de los problemas complejos utilizados en la investigación sobre RPC que permiten

diferenciarlos de las tareas tradicionales para medir inteligencia y las tareas analíticas de RP (Funke, 2001).

Además, estas características permiten diferenciar este tipo de problemas de los problemas estáticos (Herde, Wüstenberg y Greiff, 2016), por lo que en la literatura no será difícil encontrar *problemas dinámicos* para referirse a problemas complejos. Estas características relativas a las tareas utilizadas para medir la RPC, permiten operativizar la definición de un problema complejo, y son:

- (a) La complejidad de la situación.
- (b) La conectividad entre un gran número de variables que obligan a la persona a reducir la cantidad de información y anticipar efectos secundarios.
- (c) La naturaleza dinámica de la situación del problema requiere la predicción del desarrollo futuro (un tipo de planificación) y el control a largo plazo de los efectos de las decisiones.
- (d) La escasa transparencia (u opacidad) de los escenarios requiere la recopilación sistemática de información.
- (e) La existencia de múltiples objetivos requiere una cuidadosa elaboración de prioridades y equilibrar los objetivos conflictivos o contradictorios.

Greiff y Fischer (2013), relatan cómo Funke (2001) se basa en dos grandes aproximaciones o teorías en el estudio de la RPC, la aproximación funcionalista (Vollmeyer, Burns y Holyoak, 1996) y la perspectiva teórica de la acción (Dörner, 1986; citado en Greiff y Fischer, 2013) para confirmar estas cinco características.

Proceso de la RPC

El compendio de numerosas contribuciones interdisciplinarias, tanto teóricas como empíricas, han permitido establecer una teoría sobre el proceso de resolución de problemas complejos. Esta teoría, recogida en Fischer, Greiff y Funke (2012, p. 36), establece el siguiente proceso:

1. Al principio, la persona que resuelve el problema tiene que adquirir el conocimiento sobre el problema.
 - a. Se asume que la persona explora el comportamiento del sistema usando una estrategia que (a) conoce y (b) parece ser la más racional ecológicamente (ej. aleatoria o interacción sistemática con el sistema, leer las instrucciones, preguntar a un experto, etc.)
 - b. La exploración conduce a (a) el conocimiento sobre los estados del sistema y las acciones emprendidas (conocimiento de casos); como a (b) una representación interna del problema, que contiene los elementos más importantes y relaciones del sistema (conocimiento estructural), el cual es normalmente inferido desde el conocimiento de casos.
 - c. Como la capacidad de la memoria de trabajo de la persona que resuelve el problema es limitada, la representación interna es objeto de una reducción de información. Las relaciones y los elementos que proveen menor relevancia para controlar el sistema en el curso de la exploración son omitidas para permitir una planificación y previsión más eficiente.
2. Cuando la persona que resuelve el problema tiene una cierta cantidad de conocimiento sobre el problema que va a resolver, se supone que aplica el conocimiento de cara a conseguir su objetivo.

- a. La persona usa su representación interna para hacer predicciones sobre la dinámica del sistema para decidir (a) si tiene que intervenir, y (b) qué intervención tendrá consecuencias aceptables en la situación actual. Cuando la situación actual señala la intervención correcta de forma inmediata (dado el conocimiento de casos), la persona confía en su conocimiento de casos.
- b. Los procesos de supervisión detectan (a) el progreso en la resolución del problema, y (b) las implicaciones del feedback del ambiente para la representación del problema. Cuando la representación del problema demuestra no ser viable para alcanzar los objetivos a tiempo, la persona tiene que volver a la adquisición de conocimientos o cambiar los objetivos (dependiendo de factores como la importancia de los objetivos y el esfuerzo de adquirir más conocimiento).

A partir de este proceso podemos sostener que ya tenemos un marco bastante completo que incluye unas características que definen operativamente la RPC y suficiente apoyo empírico para determinar el proceso de RPC.

Sin embargo, como señalan Fischer, Greiff y Funke (2012), existen numerosas cuestiones sobre la resolución de problemas complejos que aún demandan una respuesta, como la existencia de ciertas estrategias especialmente útiles para hacer frente a problemas complejos. En esta investigación veremos hasta qué punto es beneficioso el empleo de una estrategia computacional, promovida por el pensamiento computacional, supuestamente más útil para la RPC que otras estrategias.

Hasta aquí se ha recogido y expuesto el estado actual de la investigación en el campo de la resolución de problemas complejos. A continuación, vamos a ver qué aporta el supuesto pensamiento computacional al estudio del pensamiento, y más concretamente, qué aporta a la resolución de problemas.

Capítulo 3. Aportaciones del pensamiento computacional a la RPC

El pensamiento computacional se puede definir, a partir de todo lo expuesto en el capítulo 1, como un conjunto de habilidades que forman una estrategia para abordar fácilmente problemas complejos, de manera que reformulemos un problema complejo en uno computacional y éste se pueda solucionar abordando las diferentes partes computables como pequeños subproblemas separados.

Un problema computacional se define como un mapeo entre el input y el output, entendiendo como input el estado dado del problema y como output un estado que satisfaga la especificación del input. Hasta ahora, el concepto *problema computacional* se ha definido por y para científicos de la computación, pero, atendiendo a esta definición, quien lo mapea puede ser una persona, un grupo, un software, o una combinación (Carruthers y Stege, 2013).

En este sentido, el término computación se ha definido como un proceso de pensamiento implicado en formular problemas de forma que sus soluciones puedan ser representadas como unos pasos computacionales y algoritmos (Aho, 2012).

Aproximaciones del pensamiento computacional en la resolución de problemas

En la investigación sobre la resolución de problemas complejos, algunos autores (Schoppek, 2002; Kwisthout, 2012) proponen un enfoque o marco computacional como estrategia para la mejora de la resolución de problemas.

Vemos cómo en las diferentes propuestas recogidas existen discrepancias sobre cuándo es necesario un enfoque computacional para abordar problemas complejos, si es

necesario como una estrategia para solucionar el problema o, por el contrario, es necesario durante todo el problema, incluyendo la representación del mismo.

Por una parte, Schoppek (2002) propone la necesidad de un marco computacional para tratar sistemas complejos, que engloba conocimiento sobre el input y output del problema, conocimiento de la estructura del problema y conocimiento estratégico para la resolución del problema.

El marco propuesto relaciona la complejidad del problema con el conocimiento estratégico, ya que, para problemas simples o estáticos, el conocimiento de la estructura del problema es suficiente para resolverlo, mientras que para problemas complejos un mayor conocimiento estratégico facilita la RPC.

En este caso, podríamos decir que el conocimiento estratégico se pone en marcha en la aplicación del conocimiento, el proceso de resolución de problemas o fase de control del sistema, pero no es necesario para la representación del problema, en el que sólo es necesario un conocimiento estructural para la adquisición del conocimiento.

Sin embargo, por otra parte, el marco computacional propuesto por Kwisthout (2012), incluye un modelo computacional para procesos relacionados con la adquisición del conocimiento, como son la abstracción y la representación del problema. Utilizando un modelo computacional sería más fácil gestionar la cantidad de variables consideradas igual de relevantes de cara a la representación, y se facilitaría así la resolución de problemas.

Definición operativa del PC y la resolución de problemas complejos

Para comprobar en qué fases de la resolución de problemas complejos está implicado el pensamiento computacional, o la estrategia computacional para ser más

exactos, vamos a dividir los procesos operativizados en el pensamiento computacional (CSTA y ISTE, 2011) en dos tipos de procesos.

En primer lugar, vamos a definir los procesos comunes con la resolución de problemas, es decir, las características descritas en el pensamiento computacional que son procesos clave en la resolución de problemas. De esta forma, podremos resaltar cómo los procesos habituales de la resolución de problemas pueden mejorar su efectividad al emplear una estrategia computacional, de cara a resolver problemas complejos o dinámicos.

En segundo lugar, analizaremos cada uno de los procesos que se proponen como específicos del PC, para conocer qué habilidades y técnicas de pensamiento pone en marcha este pensamiento como parte de una estrategia computacional para mejorar la resolución de problemas complejos.

Procesos compartidos con la resolución de problemas

Los procesos planteados como parte del pensamiento computacional, que son procesos base de la resolución de problemas son la abstracción; el tratamiento de datos, que incluye recopilación, análisis y representación de datos; y la secuenciación de problemas o uso de algoritmos. En esta sección nos centramos en describir qué aporta el pensamiento computacional a estos procesos propios de la resolución de problemas.

Abstracción

Mientras que para la RP, la abstracción es un proceso que sirve para extraer los elementos comunes de un problema o situación dada; en el PC la abstracción es descrita como una función que trabaja en múltiples capas, es decir, no sólo tiene que despojar una situación de sus detalles para quedarse únicamente con los puntos importantes y compartidos con otras situaciones; además debe engranar la abstracción realizada en cada

situación, cada capa, y abstraer los puntos comunes a todas éstas (Wing, 2006, 2008, 2011).

Por tanto, la abstracción para la RP es fundamental para realizar un correcto tratamiento de datos; y para el PC es imprescindible para el tratamiento de datos y, además, para entender las relaciones entre las diferentes capas del problema.

En este punto, cabe destacar de nuevo la propuesta de Kwisthout (2012) de un marco computacional para la resolución de problemas complejos en el que estudia la mejora del proceso de abstracción en la búsqueda de aspectos relevantes del problema si se computabiliza el problema, de forma que cada pequeño foco del problema pueda realizar el proceso de abstracción y detectar relevancias en cada uno de ellos. De esta forma se logra una mejor representación del problema, que facilitará la resolución del mismo.

Este marco computacional (Kwisthout, 2012) apoya la importancia del pensamiento computacional en la representación de problemas complejos ya que, como señala el autor, este marco no es necesario para problemas simples, ya que en éstos se pueden detectar fácilmente los aspectos relevantes del problema.

Tratamiento de datos: recopilación, análisis y representación

El tratamiento de datos en la RP y en el PC se puede considerar como un conjunto de procesos bastante similar entre ambos. Sin embargo, también resaltamos algunas diferencias que aporta el PC a estos procesos.

En cuanto a la recopilación de datos, en la RP está enfocado a una recopilación efectiva que permita facilitar el análisis posterior. Esta efectividad va de la mano con la memoria a largo plazo y la memoria de trabajo, que permiten recopilar y procesar la

información relacionada con el problema dado (Baddeley y Hitch, 1974; Wiley y Jarosz, 2012).

En el PC, el gran aporte de la recopilación de datos es que ésta implica manejar grandes cantidades de datos, además de que esta masiva recopilación de datos siga siendo efectiva.

Sobre el análisis de datos, la diferencia reside en dónde encontramos este proceso. Para la RP se trata de un proceso cognitivo implicado en la representación del problema, que junto a la recopilación de datos y abstracción de éstos, facilitan una representación que permita una planificación adecuada (Polson y Jeffries, 2014).

Para el PC, el análisis de datos lo encontramos igualmente en la representación del problema y además lo encontramos en la evaluación (Selby y Woollard, 2013). El PC requiere de un análisis dimensional que permita comprobar los tipos y modelos con los que trabajamos (Wing, 2006).

En cuanto a la representación de datos, en la RP sirve principalmente para crear una representación del problema, ya que nos ayuda a comprender la nueva información e integrarla en esquemas de conocimiento más generales (Markman, 2012).

En el PC, la representación de datos se refiere de forma similar a la creación de una representación del problema, de cara a plantear una estructura general que facilite el análisis de nuevos datos, facilitando la creación de una gran estructura que relacione las representaciones de los diferentes subproblemas (National Research Council, 2010).

Desde la investigación sobre la resolución de problemas complejos, se ha demostrado una correlación entre la resolución efectiva de este tipo de problemas y una mayor capacidad de procesamiento de información, que incluiría la recopilación, el análisis y la representación de datos descritos en el PC (Funke, 2001).

Estudios posteriores han corroborado estos resultados, como los realizados por Angeli y Valanides (2013). Estos investigadores han comprobado la relación existente entre el procesamiento de información, en el que podríamos englobar los procesos del tratamiento de datos aquí descritos, y la dependencia de dominio al enfrentarse a un problema complejo.

Además, los métodos utilizados para el procesamiento de información requieren el uso de ordenadores y el conocimiento de las *affordance* que ofrecen estos dispositivos, lo que está en consonancia con la idea de que el pensamiento computacional requiere conocer el estado actual de la computación y los dispositivos que la permiten.

Volviendo al estudio realizado por Angeli y Valanides (2013), en él participaban dos grupos experimentales a los que se les proponía resolver un problema complejo presentado de dos formas diferentes, es decir, se presentaban dos versiones del problema.

Un grupo experimental utilizaba métodos de procesamiento de datos independientes del campo, y otro grupo utilizaba métodos dependientes. Respecto a las formas en las que se presentaba el problema, una versión del problema incluía la información de las diferentes partes del problema ya dividida, es decir, el problema se había descompuesto en subpartes para abordarlo mejor, y otra versión del problema con toda la información del problema integrada en la presentación del mismo.

Sus resultados, a falta de más evidencias, apuntan a la necesidad de emplear métodos de procesamiento de información independientes de campo. Las personas que resolvieron el problema complejo usando métodos generales obtuvieron mejores resultados (Angeli y Valanides, 2013).

Con todo esto podemos plantear que la mejora en los resultados de resolución de problemas complejos debida al uso de métodos independientes para el procesamiento de

información, requiere de un pensamiento necesario para la resolución de problemas complejos, capaz de emplear ciertos procesos cognitivos de forma independiente al dominio. Este planteamiento vendría a decir que utilizar una estrategia computacional facilitaría la representación del problema.

Algoritmos o secuencialización

El uso de algoritmos se plantea como algo más o menos similar en ambos casos, pero que exige un grado mayor de dominio en el caso de problemas complejos, por lo que el empleo del PC nos facilitaría la algoritmización o secuencialización del proceso de resolución.

Desde la RP definen algoritmo como un método serial que permite progresar de forma incremental hasta el objetivo, garantizando la solución del problema mediante la aplicación de una serie de operadores (Novick y Bassok, 2005).

En el PC, Wing (2011) señala que el pensamiento computacional incluye pensamiento algorítmico. Define un algoritmo como una abstracción de un proceso que recibe datos, ejecuta la secuencia de éstos como una lista de pasos y genera datos para satisfacer el objetivo deseado (p.1).

Como podemos apreciar, en ambos casos se señala una lista de pasos que permite avanzar hasta satisfacer nuestro objetivo. Por tanto, en nuestra opinión, el uso de algoritmos no aporta ninguna novedad como proceso cognitivo, ya que el empleo del pensamiento algorítmico es característico de la resolución de problemas.

La computación de los problemas complejos, relacionada con el pensamiento computacional, diferenciaría la creación de algoritmos en problemas complejos de la creación de algoritmos en problemas simples, en que en el primer caso se crearían varios

algoritmos que pueden requerir ser ejecutados de forma simultánea, mientras que para problemas simples el algoritmo sería único.

Procesos específicos del PC

Si recordamos la definición inicial sobre pensamiento que proponían Carretero y Asensio (2004), este concepto designaba un conjunto de actividades u operaciones intelectuales.

Con esta definición, los procesos específicos del PC, descomposición del problema, automatización, paralelismo y simulación, no pueden ser considerados operaciones intelectuales o procesos cognitivos de pensamiento de orden superior, sino que es más correcto hablar de habilidades o técnicas empleadas (Johnson, 2000).

En este sentido, nuestra idea de pensamiento computacional se podría definir como un conjunto de habilidades, actividades o técnicas de pensamiento, que forman parte de una estrategia que se emplea en operaciones intelectuales relativas a la resolución de problemas. En concreto, facilitan la abstracción, el tratamiento de datos y la creación del algoritmo o secuenciación.

Por tanto, se supone que esta estrategia es útil para la resolución de problemas complejos. Pero no tenemos suficiente evidencia sobre si esta estrategia es útil para todo el proceso de resolución de problemas, para la fase de adquisición del conocimiento del problema, en la que entraría el tratamiento de datos y la abstracción; o para la fase de aplicación de conocimiento del problema, en la que únicamente sería útil para la creación de un algoritmo (o secuencia) para resolver el problema.

A continuación, se describen las habilidades concretas del pensamiento o estrategia computacional.

Descomposición del problema

Park, Song y Kim (2015) realizaron un estudio sobre el manejo de la carga cognitiva en personas con alta y baja puntuación en pensamiento computacional, para comprobar la influencia del PC, y por tanto, de la descomposición de problema, en el manejo de la carga cognitiva.

La teoría de la carga cognitiva (Sweller, 1988) nos permite empezar a operativizar el aprendizaje en términos de regulación de la carga cognitiva que supone un nuevo aprendizaje o la resolución de un problema.

Según Sweller (1988), existen tres tipos de carga: intrínseca, extraña y vinculada. La carga intrínseca se refiere a la cantidad de procesamiento cognitivo que necesitamos para comprender, la carga extraña se refiere a la cantidad de capacidad cognitiva que hemos empleado pero que no es necesaria y/o relevante para la comprensión, y la carga vinculada se refiere al procesamiento profundo en actividades de alto nivel como integrar conocimiento.

La regulación ideal de la carga cognitiva implica controlar la carga intrínseca, reducir la extraña y promover la vinculada. Promover esta última carga es imprescindible para conseguir la resolución de un problema (Sweller, 1988).

Los estudios realizados para comprobar la relación entre PC y carga cognitiva, apoyan el supuesto de que, a mayor PC, menor carga cognitiva (Park, Song y Kim, 2015), lo que sugiere que la descomposición del problema es un aspecto fundamental en el PC.

Respecto a la aparición en la literatura de la estrategia de descomposición en resolución de problemas, se aprecia un mayor uso de ésta en los expertos del dominio del problema (Penner y Voss, 1983), al enfrentarse en este caso a un problema complejo

como es ponerse en la posición de un ministro de agricultura para aumentar bajo nivel de productividad de los cultivos que se encuentra en los últimos años.

De hecho, la descomposición del problema ha sido y es una práctica común en la invención científica (Hoddeson, 2002), investigada en científicos de la talla de Faraday o Darwin. Estos datos indicarían que los expertos en el dominio tienen los conocimientos adecuados para saber cómo deben subdividir un problema complejo, como es la invención en ciencias, de modo que obtengan subproblemas independientes en su resolución que les permitan solucionar el problema global.

En esta línea, podríamos definir la descomposición del problema en el PC como la habilidad de dividir el problema en pequeños problemas independientes que permitan abordarlos por separado y de esta forma afrontar el problema de forma más sencilla, al reducir nuestra carga cognitiva.

Automatización

La automatización de procesos puede estar limitada al tamaño del problema (Estudillo, Bermudo, Casado, Prasad y Garcia-Orza, 2015). Los problemas de gran tamaño es difícil abordarlos de una forma automática, pero gracias a la descomposición del problema, podemos realizar computaciones que permitan abordar pequeños problemas que sean fácilmente abordables de manera directa y automática.

La automatización se consigue con la práctica de los procesos que deseamos automatizar, dicho de otra forma, la automatización es más viable cuanto mayor es la experiencia. Como recogen Kroll y Linck (2007) la automatización incluye cuatro características en la mayoría de las definiciones. Estas características son: un procesamiento más rápido, menor o nulo esfuerzo atencional, que el proceso ocurra fuera

de la consciencia y es balístico en su naturaleza, es decir, no se puede parar una vez iniciado (p.247).

Además, el uso de máquinas que computan, es decir, ordenadores y otro tipo de dispositivos que automatizan tareas, como la calculadora, hace que resulte más sencillo la tarea de automatizar. La clave aquí reside en la capacidad de la persona de conocer las tecnologías que mejor le pueden ayudar y saber aplicarlas adecuadamente a las tareas que lo demandan.

La automatización de tareas nos lleva a reducir considerablemente el tiempo en el que analizamos datos. Por ejemplo, la estadística asistida por ordenador permite analizar una cantidad abrumadora de datos que antes era impensable. Esto ha permitido avanzar a muchas disciplinas que hacen uso de la estadística: medicina, genética, economía, psicología y un largo etcétera.

Automatizar como característica del PC consistiría, por tanto, en realizar de forma directa las tareas más sencillas y en delegar a máquinas las tareas repetitivas que se pueden reformular de forma sencilla y computacional.

Paralelismo

El concepto de paralelismo, en la definición operativa de pensamiento computacional, se refiere principalmente al paralelismo entre acciones. Los lenguajes de programación trabajan con este principio, al realizar secuencias de instrucciones que suceden de forma simultánea (Brennan y Resnick, 2012).

Wing (2006, p.1) afirma que el PC implica realizar un procesamiento paralelo, algo que los expertos en PC incluyen el procesamiento paralelo en la definición de PC (NRC, 2010). Sin embargo, en la literatura revisada no se ha visto ninguna descripción

sobre cómo actúa supuestamente este procesamiento paralelo en el pensamiento computacional.

Nuestra visión sobre qué significa la implicación de un procesamiento paralelo en esta característica, a partir de cómo se interpreta esta característica en su definición operativa, reside en la necesidad de procesar simultáneamente dos acciones independientes para encontrar una relación entre ellas que pueda producir un beneficio.

Simulación

Algunas referencias a este proceso se realizan desde una perspectiva de replicar algo físico en un ambiente virtual para poder experimentar con ello de forma continua y sin riesgos, por ejemplo, simulaciones científicas (Basawapatna, Koh, Repenning, Webb y Marshall, 2011; Ioannidou, Bennett, Repenning, Koh, Basawapatna, 2011).

Sin embargo, la implicación más importante de la simulación en el pensamiento computacional se encuentra cuando la relacionamos con los complicados procesos de abstracción que se necesitan en el PC. En este sentido, debemos tener en mente la relación entre cada par de capas, definiendo esta relación a través de una función de abstracción (Wing, 2008). Necesitamos trabajar con simulación porque las soluciones a los distintos subproblemas tienen que engranar adecuadamente.

Al definir el PC, Weinberg (2015) habla de la representación de datos a través de abstracciones, como modelos y simulaciones. En este caso, la simulación es reflejada como el resultado de un proceso de abstracción.

Según Matlock (2004), las personas construimos modelos que se parecen al espacio físico, simulan objetos en estos modelos espaciales, y simulan movimientos similares en cierta medida a la percepción del movimiento físico. Esta aproximación a la

simulación, reúne características de las dos perspectivas tratadas en este punto: simular espacios físicos, y simular posibles resultados alcanzados mediante abstracción.

Existen evidencias sobre el empleo de simulaciones en la resolución de problemas, relacionadas con la representación de la estructura relacional de la situación del problema (Rosales, Vicente, Chamoso, Muñoz y Orrantia, 2012). Sin embargo, no se trata de un aspecto que esté implicado necesariamente en la resolución de problemas, por lo que se puede considerar como un posible componente de algunas estrategias de resolución utilizadas para problemas complejos.

Greitemeyer y Würz (2006) confirman esta suposición sobre que la simulación no está necesariamente implicada en la resolución de problemas, al realizar un estudio cuyos resultados arrojan que la simulación facilita lograr objetivos difíciles, pero su uso es indiferente cuando tratamos de alcanzar un objetivo fácil. En esta línea, podríamos sugerir que la simulación, como factor incluido en un pensamiento computacional, supone simular posibles resultados abstraídos del problema que pueden implicar la simulación de espacios físicos.

PC como facilitador de la RPC

Para afirmar que el pensamiento computacional facilita la resolución de problemas complejos debemos estudiar si realmente este pensamiento, o estrategia computacional, lo facilita.

Además de estudiar esta hipótesis general, resulta necesario investigar y conocer cómo este pensamiento, o estrategia de pensamiento para ser más exactos, facilita la resolución de problemas complejos, es decir, necesitamos conocer en qué fases de la RPC está implicado el pensamiento computacional, ya que no tendremos una definición del

pensamiento computacional adecuada hasta que no conozcamos cómo puede influir éste en cada parte de la resolución de problemas complejos.

Así, nuestra intención es explorar el concepto de pensamiento computacional para averiguar su alcance en la resolución de problemas y establecer una definición adecuada de este pensamiento, ya que, al igual que sugieren Blackwell, Church y Green (2008), es importante conocer las ventajas de adoptar este nuevo estilo de pensamiento en la resolución de problemas, de modo que realmente aporte valor y sepamos aprovechar sus ventajas.

Por tanto, esta tesis doctoral tiene como objetivo estudiar cómo afecta el pensamiento computacional en la RPC, de forma que, en función de los resultados que obtengamos, podamos conocer en qué fases de la resolución de problemas complejos está implicado este pensamiento.

Capítulo 4. Investigación sobre resolución de problemas complejos y pensamiento computacional

Para definir el método elegido para esta investigación, vamos a dedicar un capítulo a repasar las corrientes y las técnicas sobre la investigación en resolución de problemas, desde sus inicios hasta la actualidad, cuando podemos estudiar la resolución de problemas complejos de una forma bastante adecuada. Por otra parte, se recogen los enfoques propuestos en estos últimos años para investigar el pensamiento computacional.

Evolución en la investigación sobre resolución de problemas

Como recoge Funke (2013), la investigación sobre resolución de problemas ha aumentado en gran medida durante los últimos años, y con la llegada de nuevos métodos para el rastreo de procesos, la creación de micromundos virtuales y la definición de nuevos constructos, las teorías están siendo ajustadas.

La revisión que hace Funke (2013) de la evolución de las corrientes y técnicas en la investigación sobre RP desde la década de los 80 y de su estado actual, nos proporciona unas altas expectativas sobre la cercana llegada de un marco sólido para el estudio de la resolución de problemas, ya que se ve el gran crecimiento de estudios sobre este tema en los últimos 10 años.

Aunque la investigación sobre resolución de problemas puede considerarse bastante reciente, existen casos previos de investigaciones relacionadas con la resolución de problemas. Por ejemplo, a finales de los años 40, Berg propuso una simple técnica para medir la abstracción del pensamiento, proceso implicado en la resolución de problemas.

Esta técnica mide si la persona cambia su objetivo, para conocer la flexibilidad del pensamiento, a partir de una serie de indicadores como son el número de errores, el tiempo y la cantidad de perseverancia (Berg, 1948).

Corrientes en la investigación de resolución de problemas

Frensch y Funke (2014) recogen cómo evolucionan los objetivos de investigación en resolución de problemas a partir de los años 70. Por una parte, en Estados Unidos se empezó a investigar sobre la experiencia de dominio, intentando conocer los mecanismos generales y los mecanismos específicos de cada dominio para entender mejor cómo se produce la resolución de problemas.

Por otra parte, en Europa se empezó a investigar con problemas computarizados, pero con distintos objetivos y metodologías. Mientras que en Reino Unido, Broadbent empezó a investigar los procesos conscientes e inconscientes implicados en la RP; en Alemania, Dörner comenzó a investigar la relación de ésta con componentes sociales y motivacionales (Frensch y Funke, 2014).

El surgimiento de distintas corrientes en los años 70 se debió a que, hasta esa fecha, se habían evaluado problemas a través de tareas de laboratorio relativamente sencillas (ej. el problema de los rayos X, la Torre de Hanoi,...), ya que estas tareas tenían claramente definidas las soluciones más óptimas, eran resueltos en un corto periodo de tiempo y los pasos eran fácilmente verbalizables (Frensch y Funke, 2014).

Como señalan estos autores (Frensch y Funke, 2014), a partir de los 70 surgió una corriente que cuestionaba que estos problemas no activaban los mismos procesos cognitivos que los problemas complejos reales. Por eso, y aunque estos problemas de laboratorio fueran fáciles de medir, se produjo un intento de superación de esta cuestión.

Técnicas tradicionales de investigación en la resolución de problemas

Hasta que a finales de los años ochenta llegaron las técnicas de neuroimagen y cambiaran la aproximación a la investigación sobre resolución de problemas, ésta ha utilizado tradicionalmente tres técnicas diferentes al evaluar problemas de laboratorio

(Smith y Kosslyn, 2008). Estas técnicas son registrar la conducta, analizar el protocolo verbal y comparar la conducta humana en resolución de problemas con modelos informáticos.

El registro de la conducta se realiza de forma secuencial, registrando el movimiento de los participantes por el espacio del problema, de forma que se pueden obtener los tiempos que tardan en alcanzar el siguiente estado del problema para trazar un gráfico que muestre el proceso de resolución del problema para cada participante.

El análisis del protocolo verbal incluye la grabación de un vídeo o audio que permita transcribir todo el proceso, gracias a que los participantes piensan en voz alta, para posteriormente analizar la representación del problema y la secuencia de pasos empleada. De esta forma se deduce el espacio problema que construye cada participante.

Por último, la técnica basada en replicar la estrategia de resolución por un ordenador, pretende comparar los resultados en términos de movimientos por el espacio problema. Estos modelos permiten estudiar técnicas heurísticas de resolución de problemas como son la búsqueda aleatoria, la escalada y el análisis medios-fín.

La evolución de las corrientes y las técnicas de investigación han permitido que en la actualidad podamos investigar la RPC de un forma más fiable y válida.

Investigación de la resolución de problemas complejos

La investigación que particulariza la resolución de problemas complejos se propagó en los años 90 con las investigaciones de Sternberg y Frensch (1991), Funke y Frensch (1995), y Dörner (1997); en las que ya se utilizaban escenarios simulados por ordenador y micromundos como herramientas de evaluación de la RPC (Funke, 2013; Greiff, Fischer, Wüstenberg, Sonnleitner, Brunner y Martin, 2013).

El uso de ordenadores en la evaluación de RPC ha permitido superar la limitación de los instrumentos de evaluación de lápiz y papel, ya que se daba una falta de correspondencia entre los requisitos de la evaluación y los operadores disponibles para desarrollarla (Greiff, Fischer et al., 2013).

La posibilidad de crear actividades interactivas y que cambian de forma dinámica es la principal ventaja de los entornos computarizados. En general, como indican Greiff, Fischer et al. (2013), “se han desarrollado dos aproximaciones para evaluar la resolución de problemas complejos: micromundos y escenarios basados en marcos formales/sistemas complejos múltiples” (p. 581).

Estos instrumentos de evaluación han sido una respuesta que escapa a los clásicos escenarios de laboratorio, y que en principio permitían un alto grado de fidelidad de la realidad y permitían un control sistemático de diversos factores influyentes (Brehmer y Dörner, 1993).

Por una parte, los micromundos parecen poseer validez ecológica. Sin embargo, como apunta Funke (2001), tienen dos limitaciones:

1. No queda clara la diferencia entre los diferentes escenarios, es decir, si existen diferencias más allá de las características superficiales como la semántica;
2. No queda claro cómo comparar la mayoría de los sistemas en base a un marco formal, a una estructura subyacente.

Ambas limitaciones dificultan el hecho de que podamos estar seguros de que estamos midiendo realmente el constructo de resolución de problemas complejos, y por tanto necesitamos pruebas que se ajusten al constructo a evaluar.

Como señalan Ercikan y Oliveri (2016) la evaluación de constructos complejos requiere una cuidadosa planificación, y estos autores destacan 3 aspectos esenciales para la investigación de constructos complejos.

El primero de ellos es tener en cuenta la complejidad del constructo a lo largo de las diferentes etapas del desarrollo de la investigación, tales como el diseño, la escalabilidad y los aspectos interpretativos. El segundo asunto esencial se refiere a conseguir evidencias sobre la validez de las pruebas más allá de los análisis psicométricos tradicionales de patrones de respuesta. Por último, se ha de tener en cuenta la validez transcultural (Ercikan y Oliveri, 2016).

Por otra parte, en la creación de marcos formales destaca Funke (2001), quien establece un marco formal que pretende conseguir una investigación sistemática de la RPC. Este marco se basa en las características que definen el constructo de qué es un problema complejo desde la perspectiva europea de investigación de RPC.

Uso de marcos formales para el estudio de la RPC

La introducción de marcos formales en la evaluación ha permitido aumentar la escalabilidad y las propiedades psicométricas de la evaluación de RPC, que era un problema propio del uso de micromundos.

Funke (2001) propuso dos marcos: los sistemas lineales de ecuaciones estructurales (LSE en inglés) y los autómatas de estados finitos (FSA en inglés).

Ambos marcos miden el proceso de la resolución de problemas, al contrario que las pruebas de inteligencia que miden el resultado. Mientras que los sistemas lineales de ecuaciones estructurales (LSE) miden la relación entre variables cuantitativas, los autómatas de estados finitos (FSA) miden las relaciones entre variables cualitativas (Funke, 2001).

Como describe Funke (2001), en las ecuaciones lineales estructurales (LSE) existen variables exógenas, que pueden ser manipuladas por el sujeto, y variables endógenas, que no pueden ser manipuladas directamente. La tarea general en las LSE implica (a) encontrar cómo están relacionadas las variables exógenas y endógenas entre ellas, y (b) controlar las variables en el sistema que alcanzan cierto valor del objetivo. Como vemos, la identificación del sistema y el control del sistema se han separado como dos pasos de la tarea.

Por su parte, los autómatas de estados finitos (FSA) se caracterizan por tres cualidades que son (a) sólo pueden tener un número finito de estados, (b) desde el estado inicial ellos se mueven al siguiente estado a través de un input o entrada como pulsar una tecla o a través de un proceso autónomo, y (c) normalmente, el output o salida depende del input o entrada de la persona y del estado alcanzado (Funke, 2001). Como vemos, al igual que en los LSE, la identificación del sistema y el control del sistema se han separado como dos pasos de la tarea.

Características de la RPC y los marcos formales

La resolución de problemas complejos se ha descompuesto habitualmente en dos fases: adquisición de conocimiento sobre la tarea, la representación del problema, y aplicación del conocimiento, el proceso de resolución del problema (Novick y Bassok, 2005).

Esta descomposición ha permitido profundizar más en las dos dimensiones definidas. De esta forma, los problemas desarrollados para medir la RPC delimitan ambas dimensiones para analizar los resultados de forma independiente y luego correlacionarlos. En la literatura sobre sistemas dinámicos o problemas complejos, la adquisición del conocimiento se denomina identificación del sistema, y la aplicación del conocimiento también se denomina control del sistema.

Como apunta Funke (2001), la *adquisición de conocimiento*, o lo que es lo mismo, la identificación del sistema, describe una situación compleja de aprendizaje de cara a encontrar los detalles sobre la conectividad de las variables y su dinámica.

En la situación de LSE la identificación del problema requiere una estrategia de identificación, es decir, saber cómo manipular las variables exógenas y poder derivar la estructura causal del sistema de las consecuencias de esta manipulación, además la relación del sistema puede ocurrir en diferentes niveles como la dirección, aspectos cualitativos (positivo o negativo) de la relación, etc.

En la situación de FSA la tarea es similar porque los efectos de las entradas en las salidas y en los estados del sistema tienen que ser descubiertos, pero difiere en que en este caso la tarea consiste en detectar condiciones que debemos satisfacer de cara a hacer la transición entre los estados.

En la *aplicación del conocimiento*, o control del sistema, se describe la situación de aplicar el conocimiento adquirido previamente para alcanzar un cierto estado objetivo del sistema.

En la situación de LSE se requieren dos subobjetivos: por una parte, transformar un estado dado en un estado más cerca del objetivo, transformando una variable exógena que produzca una modificación en las variables endógenas; y por otra parte, mantener este estado en un nivel estable porque el estado objetivo podría desaparecer al tratarse de un sistema dinámico.

En la situación de FSA requiere encontrar un camino desde el estado inicial al objetivo, siendo una tarea bastante fácil que puede convertirse en una tarea difícil si se dan una serie de condiciones previas que tienen que cumplirse para que se pueda dar esta secuencia o camino (Funke, 2001).

Más recientemente Greiff, Wüstenberg y Funke (2012), han propuesto un enfoque de *sistemas complejos múltiples* (MCS en inglés) que permite aumentar la complejidad de estos dos marcos formales. Para ello, este enfoque se apoya en el uso sistemas informáticos que permitan reproducir y medir la interacción entre el problema y la persona, de una forma más sencilla y fiable que las pruebas de lápiz y papel.

Su propuesta consiste en introducir un conjunto de pruebas con situaciones diferentes e independientes, que pueden ser resueltas en 5 minutos en lugar de 45 minutos (Greiff, Fischer et al., 2013). Este enfoque se ha incorporado parcialmente en las tareas utilizadas para la evaluación de la RPC en el programa PISA 2012 (Herde, Wüstenberg y Greiff, 2016).

Estos autores resumen las tres ventajas principales del enfoque de sistemas complejos múltiples y son: (1) permite discriminar varios niveles de competencia, (2) estima con una alta fiabilidad la adquisición del conocimiento y la aplicación del conocimiento, y (3) limita dificultades en las puntuaciones finales de RPC debidas a errores aleatorios aislados al comienzo de la evaluación (Herde, Wüstenberg y Greiff, 2016).

Problemas utilizados en la RPC

Problemas basados en los marcos formales

La cooperación entre disciplinas de psicología y educación ha permitido elaborar aproximaciones de evaluación con bastante éxito, de forma que las medidas psicométricas de las pruebas utilizadas en las pruebas PISA son favorables (Greiff, Holt y Funke, 2013), que recordemos están basadas en los marcos formales propuestos por Funke (2001), en los que se aplican parcialmente los MCS (Greiff, Wüstenberg y Funke, 2012).

Entre las pruebas más relevantes en la investigación de problemas complejos dinámicos basados en marcos formales, destacan MicroDYN y GeneticsLAB, basadas en el marco de sistemas lineales de ecuaciones estructurales (LSE), y MicroFIN, basada en el marco de sistemas autómatas de estados finitos (FSA). Estas pruebas han demostrado tener convergencia suficiente para guiar el camino de la experimentación en el campo de la resolución de problemas complejos dinámicos (Greiff, Fischer et al., 2013).

Como podemos comprobar en la tabla 1, elaborada por Greiff y Fischer (2013), cada una de las diferentes fases que requiere las pruebas MicroDYN, mantienen una relación directa con cada una de las características que definen los problemas complejos.

Tabla 1. Las 5 fases de MicroDYN y su relación con las 5 características de la RPC. Fuente Greiff y Fischer, 2013, p.50

Cognitive process	Characteristic feature	MicroDYN phase
Representation	Intransparency of the situation	Information generation
	Complexity of the structure	Information reduction
	Interconnectedness of variables	Model building
Solution	Polytely of the task	Goal elaboration and balancing
	Dynamics of the system	Prognosing, planning and decision making

A continuación, describimos brevemente las tareas utilizadas para comprobar su convergencia, y posteriormente comparamos el proceso de las tres pruebas para extraer los aspectos del diseño y el procedimiento comunes a las tres pruebas.

MicroDYN (LSE)

El marco MicroDYN ha favorecido una evaluación válida y motivada de la resolución de problemas complejos, proporcionando diferentes indicadores de resultado (Greiff, Wüstenberg, Holt, Goldhammer y Funke, 2013).

Esta prueba consta de diferentes tareas como la de “Entrenamiento de Balonmano”, utilizada en la evaluación PISA (OCDE, 2013).

Como se puede ver en la figura 4, a la izquierda se muestran las combinaciones de las variables de entrada (ejercicios 1, 2 y 3) desde "--" a "++". A la derecha se muestran los valores de las variables de salida (motivación, potencia de tiro, agotamiento), tanto los valores actuales en una puntuación concreta (p.e. motivación: 21) y los puntos representados en la gráfica, como los valores objetivos en rangos (p.e. motivación: 20-22) y con una línea roja en la gráfica. Por último, abajo aparece el modelo de resolución correcto.

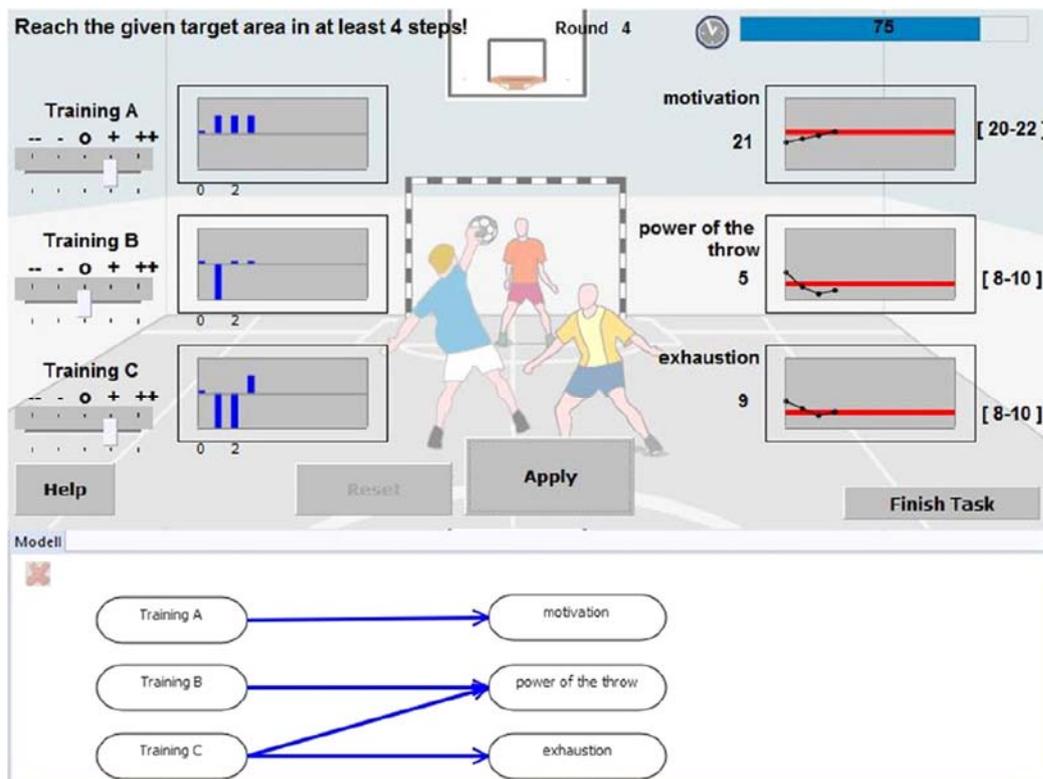


Figura 4. Impresión de pantalla de la tarea “Entrenamiento de balonmano” de MicroDYN. Fuente: Wüstenberg et al., (2012)

La aplicación de esta prueba está pensada para hacerse en cuatro minutos y medio, tres minutos para la primera fase y un minuto y medio para la segunda fase. Greiff, Fischer et al (2013) recogen una descripción detallada de esta prueba.

Genetics Lab (LSE)

Esta prueba consta de diferentes tareas sobre influencia genética. En la tarea que aparece en la figura 5, los examinados tienen que manipular los valores de las variables de entrada que aparecen en rojo a la izquierda de la pantalla. Estas variables representan los genes de las criaturas ficticias. En función de su manipulación, cambiarán los valores de las variables de salida. Estas variables de salida corresponden a las características físicas de estas criaturas. A la derecha de la pantalla, aparecen en verde los diagramas causales que los examinados se pueden ir creando con el conocimiento adquirido.

En estas pruebas no influye el conocimiento previo sobre genética, ya que no existe conexión entre los efectos de manipular los genes que aparecen en las pruebas con sus efectos en la vida real (Greiff, Fischer et al, 2013).

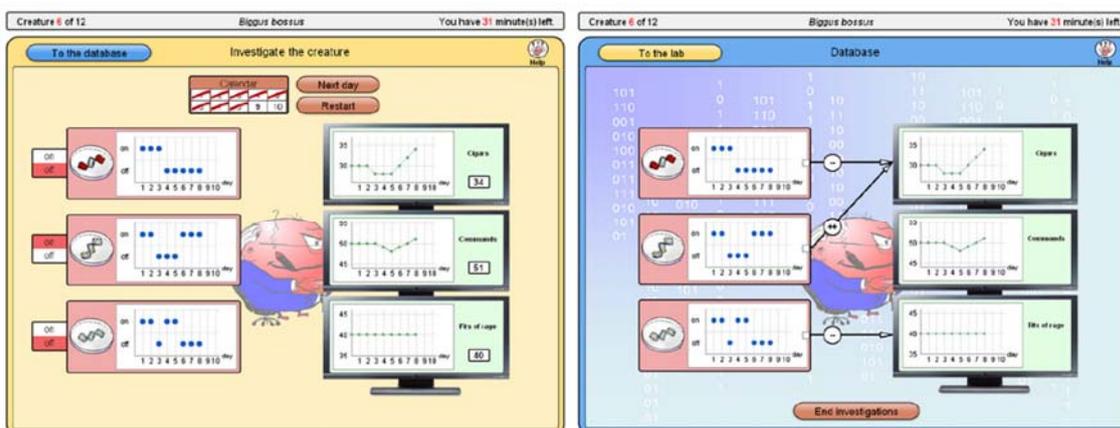


Figura 5. Impresión de pantalla de una tarea de Genetics Lab. Fuente: Greiff, Fischer et al. (2013)

La aplicación de las doce tareas de Genetics Lab está pensada para hacer en un total de 35 minutos.

MicroFIN (FSA)

A diferencia de los instrumentos anteriores, en MicroFIN nos encontramos ante un problema con estados finitos, es decir, con una estructura formal que tiene cambios de estado cualitativos.

Por ejemplo, en la tarea de la mascota debemos solucionar diversos estados de nuestra mascota, como hambre o aburrimiento, mediante diferentes botones. Otro ejemplo es la tarea social en la que hay dos personas y varios objetivos, y debemos donar a los otros los objetos adecuados según nuestro estado de ánimo y el estado de ánimo de los otros.

En la figura 6, podemos observar la tarea de la televisión en la que, en función de los estados en los que se encuentra la televisión en ese momento (mostrado en la pantalla de la derecha), debemos guiar al mando a distancia mostrado en la izquierda hacia un objetivo concreto con la menor cantidad de pasos posibles.



Figura 6. Impresión de pantalla de la tarea de “televisión” de MicroFIN. Fuente: Greiff, Fischer et al. (2013)

La aplicación de esta prueba está planteada para que la primera fase se haga en tres minutos y las pruebas de la segunda fase en un minuto cada una.

Aspectos comunes en las pruebas

Todas estas pruebas se diseñan en torno a dos fases de implementación, que ayudan a diferenciar y profundizar en la evaluación de las dos fases de resolución de problemas, conocimiento y control (Novick y Bassok, 2005).

Por tanto, en la primera fase todas estas pruebas pretenden medir el conocimiento sobre las relaciones causales entre las variables de entrada y de salida. La diferencia principal recae entre las pruebas de LSE y la prueba de FSA, ya que las primeras miden relaciones causales cuantitativas, mientras que la segunda mide las conexiones cualitativas entre las variables y los diferentes estados discretos.

En la segunda fase las tres pruebas miden el control del sistema subyacente. Deben guiar unas acciones hacia la consecución de un objetivo, con la menor cantidad de pasos posible.

El procedimiento de las pruebas también es similar. En la primera fase se permite una libre exploración de la tarea para detectar las variables y sus relaciones. En la segunda fase, se les propone un objetivo para que apliquen su conocimiento para lograrlo, preguntando por los valores “objetivo” de las variables de salida.

Traveling Salesman Problem

El problema del viajante de comercio (Traveling Salesman Problem o TSP, en inglés) es un problema de optimización combinatoria en el que debemos encontrar la ruta menos costosa, pudiendo demandar el menor tiempo, la menor distancia, la más barata, ... en sus diferentes versiones. La ruta siempre debe pasar por una serie de ciudades o

nodos, y volver a la dirección de partida, conociendo los costes fijos entre cada par de nodos (Mueller, Perelman, Tan y Thanasuan, 2015).

Este problema es un tipo de problema matemático que no ha podido ser tratado de forma computacional, lo que significa que no hay un procedimiento algorítmico general que garantice resolverlo en el menor tiempo, y aunque haya procedimientos heurísticos que permitan encontrar buenas soluciones, ninguna es la solución óptima (Applegate, Bixby, Chvatal y Cook, 2011).

El estudio de la existencia de diferencias individuales con este problema, recordemos no abordable de forma computacional, ha demostrado la influencia de la inteligencia fluida, medida como habilidad de razonamiento abstracto y habilidades analíticas, y la influencia de la inteligencia viso espacial, medida como la habilidad de manipular representaciones mentales de formas, figuras y posiciones de objetos (Chronicle, MacGregor, Lee, Ormerod y Hughes, 2008). Esta influencia era mayor cuanto mayor era la complejidad del problema.

Estas evidencias apoyarían la hipótesis de que los procesos de abstracción, tratamiento de la información y creación de algoritmos están siempre presentes en la resolución de problemas, y cuanto más complejo es éste, se necesitarían mayores niveles de eficacia en el empleo de estos procesos.

Estos resultados harían replantearse la necesidad de considerar todos los procesos cognitivos que se han definido en el pensamiento computacional. Ya que el nivel de empleo de los procesos compartidos entre el PC y la RP, influirían inevitablemente en la resolución de problemas complejos, aunque no utilizemos el resto de los procesos cognitivos, los procesos propios, del pensamiento computacional para abordarlo.

Investigación sobre pensamiento computacional

Brennan y Resnick (2012) proponen un marco teórico para estudiar y evaluar el desarrollo del pensamiento computacional. Este marco está dividido en tres dimensiones clave que facilitan la evaluación del pensamiento computacional. Sin embargo, la evaluación que permite este marco está más relacionada con el aprendizaje de técnicas para programar en Scratch que con el desarrollo del pensamiento computacional.

En primer lugar, se encuentran los conceptos computacionales, que según Brennan y Resnick (2012) son los conceptos que deben ocuparse los diseñadores a medida que programan. Como vemos, una vez más se confunden los procesos cognitivos del pensamiento computacional con técnicas usadas en programación. Estos conceptos incluyen: secuencias, ciclos, eventos, paralelismo, condicionales, operadores, datos (como variables y listas). Únicamente podríamos incluir las secuencias (creación de algoritmo), el paralelismo y los datos (en el sentido más abstracto, ya que su evaluación se ciñe a medir la creación de variables y listas en el programa de Scratch).

En segundo lugar, el marco propuesto por Brennan y Resnick (2012) incluye prácticas del pensamiento computacional o prácticas computacionales, que se enfocan en el proceso de pensar y aprender, y van más allá del qué se aprende para centrarse en el cómo. En este apartado deberían incluir las técnicas de pensamiento propias del PC y algunas cuestiones relacionadas con la aplicación del conocimiento. En esta dimensión se incluye el ser incremental e iterativo, ensayar y depurar, reusar y remezclar, y abstraer y modularizar. Estas prácticas pueden ser consideradas como parte del proceso de resolución de problemas complejos en general, y la única práctica que podríamos considerar como parte del pensamiento computacional sería el modularizar.

Por último, estos autores incluyen la dimensión de perspectivas del pensamiento computacional, que pretende describir los cambios de perspectiva y apreciación. Las perspectivas que se pueden dar en este marco son: expresar, conectar y preguntar.

Todo este marco está propuesto en base al entorno gráfico de programación de animaciones y videojuegos Scratch, creado por Michael Resnick. Por tanto, su autor se basa en su propio entorno para intentar estudiar y evaluar los aspectos del pensamiento computacional.

Sin embargo, Brennan y Resnick (2012) no han identificado correctamente los aspectos del pensamiento computacional que le pueden hacer ser un pensamiento diferente y necesario, ya que los aspectos evaluados no se corresponden en grandes rasgos a la resolución de problemas y a habilidades propias de las CC, sino que realiza una evaluación del aprendizaje de un entorno muy concreto como es Scratch.

En este mismo artículo Brennan y Resnick (2012) proponen diversos enfoques de evaluación que proponen el cómo evaluar en educación estas dimensiones, a través de diversas actividades de aprendizaje a través del diseño, que a su vez sirven para evaluar los aspectos incluidos en cada dimensión.

Aunque, como hemos dicho, no compartimos que los aspectos incluidos en el marco que han propuesto evalúen el pensamiento computacional, las actividades mediante las que evalúan pueden ser útiles para aprender programación de una forma creativa y no repetitiva, lo que sí estaría relacionado con el pensamiento computacional, abordar problemas (Zapata-Ros, 2015). En la figura 7, podemos ver una tabla resumen de qué ventajas e inconvenientes atañen a cada actividad de diseño propuesta.

	Concepts	Practices	Perspectives
<i>Approach #1: Project Analysis</i>	presence of blocks indicates conceptual encounters	N/A	N/A (possibly by extending analysis to include other website data, like comments)
<i>Approach #2: Artifact-Based Interviews</i>	nuances of conceptual understanding, but with limited set of projects	yes, based on own authentic design experiences, but subject to limitations of memory	maybe, but hard to ask directly
<i>Approach #3: Design Scenarios</i>	nuances and range of conceptual understanding, but externally selected projects	yes, in real-time and in a novel situation, but externally selected projects	maybe, but hard to ask directly

Figura 7. Fortalezas y limitaciones de los enfoques de evaluación, correspondientes a actividades de diseño: análisis de proyectos, entrevistas basadas en artefactos y diseño de escenarios. (cf. Brennan y Resnick, 2012)

En base al marco de Brennan y Resnick (2012), que diferencia conceptos, prácticas y perspectivas dentro del pensamiento computacional, Lye y Koh (2014) analizan los instrumentos de intervención en 27 artículos que desarrollan y evalúan el pensamiento computacional. En su revisión, descubren que sólo 6 de las 27 investigaciones evalúan las prácticas del pensamiento computacional, y que únicamente 2 incluyen en la evaluación las perspectivas del PC.

Desde entonces, han surgido enfoques diferentes para evaluar el pensamiento computacional, casi todos ellos con una aproximación desde la educación (ver, p.e., Koh, Nickerson, Basawapatna y Repenning, 2014; Moreno-León, Robles y Román-González, 2015).

El enfoque propuesto por Koh, Basawapatna, Bennett, Repenning (2010), y posteriormente por Koh, Nickerson, Basawapatna y Repenning (2014), incluye nueve patrones relativos al pensamiento computacional: control de usuario, generación, absorción, colisión, transporte, empujar, tirar, difusión y escalada.

Estos patrones están diseñados para evaluar el significado semántico de los juegos y simulaciones presentadas a los estudiantes. El análisis que ellos realizan detecta qué patrones de pensamiento computacional se aplican en una presentación dada. Sin embargo, los patrones descritos no se corresponden con procesos cognitivos sino que más bien son técnicas de programación.

Otro enfoque que recoge diversos aspectos del PC es el propuesto por Moreno-León, Robles y Román-González (2015), con Dr. Scratch. Rechazamos este enfoque principalmente por dos motivos: en primer lugar, propone abstracción y descomposición de problemas como un único aspecto, y en segundo lugar, incluye aspectos más propios de la programación que del pensamiento computacional.

El proponer abstracción y descomposición de problemas como un aspecto en lugar de como dos aspectos diferentes, pone en peligro la evaluación de ambos procesos cuando ocurran de forma independiente. No es lo mismo abstraer una idea a partir de un caso concreto, que abordar un gran problema en pequeñas partes.

Por otra parte, mientras que abstracción, descomposición de problemas, paralelismo, pensamiento lógico (como análisis de información), control de flujo (como creación de algoritmos) y representación de la información sí que están presentes entre los procesos cognitivos que se han operativizado como parte del pensamiento computacional (CSTA y ISTE, 2011); sincronización e interactividad con el usuario no son procesos cognitivos, sino que más bien se trata de técnicas propias de programación.

El uso habitual de la programación como herramienta para desarrollar el pensamiento computacional, ha propiciado que estos enfoques admitan técnicas de programación como procesos cognitivos. Cabe destacar también el poco tiempo que tienen estos enfoques, que indudablemente irán mejorando y ajustando sus pruebas para la medición del PC.

Una aportación reciente para la evaluación del PC ha surgido de la mano de Park, Song y Kim (2015), quienes analizan la habilidad de resolución de problemas y la carga cognitiva en relación con el pensamiento computacional, mediante la técnica de electroencefalograma (EEG) (figura 8). Estos autores se basan en los procesos que recoge la definición operativa del PC propuesta por CSTA e ISTE (2011).

Concluyen que el PC ayuda en la capacidad estratégica para situaciones problemáticas complejas, ya que el grupo que había sido formado en pensamiento computacional seguía un proceso en el que mostraban una gran actividad cerebral estable y rápidamente volvían a un estado más relajado para proceder a la resolución. Por el contrario, el grupo que no había sido entrenado en PC seguía un proceso más irregular de actividad cerebral y mayor carga cognitiva que la media (Park, Song y Kim, 2015).



Figura 8. Resultados sobre la carga cognitiva en el grupo entrenado en pensamiento computacional (CT-group en azul) y en el grupo no entrenado en pensamiento computacional (NCT-group en gris). Fuente Park, Song y Kim (2015)

La información recogida en este capítulo sobre la investigación en resolución de problemas complejos y la investigación en pensamiento computacional, nos ha permitido decidir el marco que vamos a emplear para estudiar nuestra hipótesis de investigación.

En base a los estudios aquí expuestos sobre RPC, consideramos más adecuado medir la resolución de problemas complejos utilizando un marco formal, concretamente el MCS de Greiff, Wüstenberg y Funke (2012), que permite establecer varios niveles y fases, pudiendo profundizar en su evaluación.

Los estudios sobre PC sugieren que todavía no hay un marco de evaluación consistente, por lo que vamos a considerar las diferencias en la variable PC en función de la asistencia a un curso de programación enfocado a comprender y aplicar habilidades computacionales en los procesos de resolución de problemas complejos, como veremos en los siguientes capítulos.

Sin duda, la enseñanza del pensamiento es una cuestión que se viene estudiando con mayor interés desde hace unos cuarenta años. Como señalan Ritchhart y Perkins (2005) existen una variedad de programas para enseñar a pensar que abordan diferentes aspectos como el pensamiento creativo, el pensamiento crítico, la resolución de problemas, la metacognición, etc., pero todavía nos encontramos en la superficie del conocimiento sobre cómo abordar la enseñanza de pensamiento.

Parece claro que, para enseñar un pensamiento, debemos centrarnos primero en que los alumnos aprendan a pensar para aprender, es decir, comprendan qué metas tienen que alcanzar para desarrollar un pensamiento, entrando en estrategias metacognitivas. Sin embargo, como señalan Ritchhart y Perkins (2005), la enseñanza del pensamiento necesita todavía abordar la gran cuestión sobre cómo integrar esta enseñanza en otras prácticas, en el colegio y fuera del colegio, en un camino efectivo.

En esta investigación, se pretende comprobar cómo se puede desarrollar el pensamiento computacional a través de la programación, ya que parece ser la práctica con mayor relación, y por tanto su enseñanza resultaría más natural a través de este recurso.

Bloque 2. Estudio empírico

Capítulo 5. Método de la investigación

Diseño experimental e hipótesis de estudio

Esta investigación tiene un diseño cuasi-experimental, ya que la pertenencia al grupo experimental no puede ser asignada de forma aleatoria para la investigación, pero se ha podido utilizar un procedimiento en la recogida de datos con gran similitud con un experimento. Se han podido diferenciar dos grupos en la variable independiente para conocer su influencia sobre la variable dependiente.

La variable independiente es el pensamiento computacional. El pensamiento computacional que, tal y como se ha descrito en el marco teórico de la investigación, es un pensamiento que emplean habitualmente los científicos de la computación basado en la forma en la que se enfrentan a problemas y a la creación de sistemas complejos.

La variable dependiente, sobre la que queremos conocer la influencia de emplear el pensamiento computacional, es la resolución de problemas complejos, dada la supuesta relación que se ha establecido entre ambas pero que no ha sido confirmada a nivel experimental.

La investigación adopta un enfoque pre-post con grupo de cuasi control, que pretende comprobar si existen diferencias significativas entre ambos grupos, antes y después de la formación en pensamiento computacional.

Se miden si existen diferencias de forma previa a la formación para descartar que las diferencias en la capacidad para resolver problemas complejos no se deban a la formación en pensamiento computacional, sino a diferencias ya existentes.

Al tratarse de una investigación en un área concreta en la que no se conocen trabajos previos, esto es, en la relación entre pensamiento computacional y resolución de

problemas, se van a estudiar todas las relaciones posibles para determinar de qué forma se relacionan ambas variables.

Con esto, pese a la carencia de investigaciones previas, se han establecido una serie de hipótesis de partida en base a las afirmaciones que defienden la supuesta relación entre pensamiento computacional y resolución de problemas complejos.

Esta serie de hipótesis nos sirve para enmarcar los análisis que se realizarán, de cara a ofrecer un conocimiento suficientemente amplio sobre cómo influye este pensamiento en la capacidad para resolver problemas complejos y en cada una de sus fases. Las hipótesis establecidas son las siguientes:

1. El pensamiento computacional facilita la resolución de problemas complejos.
 - 1.1. El pensamiento computacional es especialmente útil para los problemas complejos de mayor nivel de dificultad.
 - 1.2. El curso de pensamiento computacional ha supuesto un desarrollo real en la capacidad de resolución de problemas complejos.
2. El pensamiento computacional facilita la adquisición de conocimiento sobre el sistema, que incluye tanto la exploración y entendimiento de los elementos del problema, como la representación y formulación del mismo.
 - 2.1. El pensamiento computacional facilita especialmente la formulación.
3. El pensamiento computacional facilita el proceso de resolución del problema, que incluye tanto la planificación y ejecución de planes para la resolución del problema, como la observación y reflexión sobre si los planes ejecutados están acercándonos a la solución del problema.
 - 3.1. El pensamiento computacional facilita especialmente la planificación.

Inicialmente sólo se defendía la relación que existía entre este pensamiento y la planificación para la resolución de problemas, en la que el pensamiento computacional ayudaría a descomponer las acciones que se deben llevar a cabo para resolver el problema y a automatizar su resolución, con recursos como el paralelismo entre acciones y la simulación de posibles resultados antes de ejecutar el plan para determinar si el camino es el adecuado para resolverlo.

Posteriormente, se incorporó en la definición de pensamiento computacional una relación con la representación del problema además de relacionarlo con la planificación de su resolución. Esta idea se fundamenta en que, si se representa el problema de forma computacional, y se aíslan las variables para estudiarlas de una en una, se obtendrá una mejor representación, lo que influirá inevitablemente en una mejor resolución del mismo.

De esta evolución en la definición del pensamiento computacional, se obtienen las hipótesis que se plantean para la investigación, ya que se requiere comprobar si hay una relación en términos generales entre ambos, pensamiento computacional y resolución de problemas complejos. O si, por el contrario, la relación existente entre ambos se localiza única y exclusivamente en determinadas fases de la resolución de problemas.

En cualquier caso, nuestra intención con esta investigación es conocer la relación que existe entre estos conceptos, para determinar la importancia de incluir la enseñanza y práctica de las estrategias de este pensamiento en la educación obligatoria. Y posiblemente abrir una nueva área de investigación en torno a su relación, y a cómo diseñar programas educativos de programación y pensamiento computacional efectivos para que los alumnos sean capaces de mejorar su capacidad para resolver problemas complejos.

Participantes

Los participantes de la investigación son 38 alumnos de 4º de Educación Secundaria de la Escuela Libre Micael situada en el municipio de Las Rozas, en la Comunidad Autónoma de Madrid, España.

Del total de los participantes, 21 de ellos han asistido a la extraescolar de programación, donde se incluyó el curso de pensamiento computacional. Por tanto, 21 de los 38 participantes pertenecen al grupo experimental, el 55,3% de la muestra total.

El otro 44,7% de la muestra, lo que equivale a 17 alumnos, no han participado en dicha extraescolar, por lo que estos alumnos pertenecerán a nuestro grupo de control.

De los participantes del grupo de control, 4 de ellos se incorporaron una semana después de que se aplicase la fase previa de la prueba, es decir, después de que se aplicase el pre-test. Esto se debió a que durante el primer semestre estuvieron de intercambio en centros educativos fuera de España.

A estos 4 participantes se les aplicó la prueba una semana después de aplicársela al resto de los participantes, considerando que no había transcurrido tiempo suficiente desde la aplicación como para tener que desechar a estos alumnos como sujetos de la muestra.

Como dato descriptivo de la muestra, se han estudiado las diferencias entre sexos en la muestra total y en los grupos. Del total de los 38 alumnos, 20 de ellos son chicos y 18 chicas. Pero si hacemos una distinción entre los chicos y chicas que pertenecen al grupo experimental y los chicos y las chicas que pertenecen al grupo control, obtenemos que en el grupo experimental hay 15 chicos frente a 6 chicas; mientras que en el grupo control hay, por tanto, 5 chicos y 12 chicas.

Instrumentos y materiales

Para esta investigación se ha desarrollado un curso para desarrollar el pensamiento computacional, basado en la propuesta de Marais y Bradshaw (2015), como material para aplicar a los participantes del grupo experimental que ha participado en la extraescolar de programación. Este curso de pensamiento computacional nos permite diferenciar los grupos de la variable independiente.

Además, se han desarrollado unas pruebas de resolución de problemas como medida de la variable dependiente, basadas en el enfoque de sistemas complejos múltiples propuesto por Greiff, Wüstenberg y Funke (2012). Este enfoque se utilizó en el programa internacional de evaluación a estudiantes PISA (OCDE, 2010). Las pruebas de resolución de problemas nos permitirán concluir si existen diferencias en la variable dependiente, entre los grupos creados en función de la variable independiente.

Pruebas de evaluación para la resolución de problemas

Para evaluar la resolución de problemas, vamos a utilizar un total de seis pruebas basadas en el enfoque de *sistemas complejos múltiples* (MCS en inglés), que aúna los marcos MicroDYN y MicroFIN, y que permite aumentar la complejidad de estos dos marcos formales (Greiff, Wüstenberg y Funke, 2012).

Dos de las pruebas incluidas, se han extraído del programa de evaluación PISA. Estas pruebas son *Climatizador* y *Billetes*, la primera relativa al marco de LSE, al que pertenece MicroDYN; y la segunda relativa al marco de FSA, al que pertenece MicroFIN.

Las cuatro pruebas restantes se han elaborado para el estudio, y se han sometido a pruebas psicométricas en un estudio piloto, que se llevó a cabo con una muestra de 66 estudiantes de 15 años. Los 38 estudiantes de la investigación con los datos del pretest, y

28 estudiantes de 4º de Educación Secundaria del Colegio Hélicon de Valdemoro, Madrid, España.

Estas pruebas han sido sometidas a una evaluación psicométrica, recogida en el capítulo 5, cuyos resultados han permitido la creación de una batería de pruebas para el conjunto de la mayoría de las preguntas diseñadas para las seis pruebas, como veremos como resultado de dicho capítulo. En este capítulo se van a describir todas las preguntas.

Tres de estas pruebas tienen un enfoque cualitativo, relativo por tanto al enfoque de MicroFIN, y son *Vídeo*, *Agenda* e *Tren*. La cuarta prueba tiene un enfoque cuantitativo, basado en MicroDYN, y es la prueba *Coche*.

Cada una de las pruebas consta de varias preguntas, por lo que una misma prueba suele incluir diferentes fases de resolución de problemas. Por tanto, aunque haya tres pruebas de un enfoque y sólo una del otro enfoque, en total hay 6 preguntas del enfoque MicroFIN frente a 4 preguntas del enfoque MicroDYN, lo que reduce el posible desequilibrio que podría provocar si existiesen diferencias en los resultados de ambos enfoques.

Además, si sumamos las preguntas de las pruebas extraídas de PISA, serían un total de 9 preguntas del enfoque MicroFIN frente a 6 preguntas del enfoque MicroDYN. A continuación, se muestra el número de preguntas por prueba.

1. Prueba *Climatizador* (PISA). 2 preguntas.
2. Prueba *Billetes* (PISA). 3 preguntas.
3. Prueba *Vídeo*. 2 preguntas.
4. Prueba *Agenda*. 2 preguntas.
5. Prueba *Tren*. 2 preguntas.
6. Prueba *Coche*. 4 preguntas.

En función del orden de aplicación de las pruebas, éstas son las siguientes:

Prueba AGENDA

Esta prueba pertenece al enfoque MicroFIN, y tiene un contexto no tecnológico y personal. El enunciado de la prueba se puede ver en la figura 1.

Esta prueba consiste en crear un calendario de estudio para los exámenes de 5 asignaturas. Por una parte, el objetivo consiste en crear todas las pegatinas necesarias para el calendario para estudiar todos los temas de cada asignatura y repasar el temario (representar y formular). Por otra parte, el objetivo será rellenar un calendario con todas las pegatinas creadas para crear el calendario de estudio (planear y ejecutar).

Pregunta 1.

Esta pregunta es de representar y formular, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 2 y 3, según la puntuación obtenida en la respuesta. El enunciado de la pregunta se puede ver en la figura 9.

Este mes tienes los 5 exámenes finales de tus asignaturas. En cada separador de colores hay una asignatura. Tienes que pinchar en cada color para comprobar los temas de cada asignatura.

Pregunta 1
Vas a hacerte un calendario para saber qué estudiar cada día. Sabes que te estudias cada tema en lo que llamas "una **PORCIÓN DE TIEMPO**", que equivale a una mañana o una tarde.

Para rellenar el calendario tienes pegatinas de los colores de cada asignatura, así que vas a **marcar cada tema de cada asignatura con una pegatina de su color**. Además, tienes que repasarlos, es decir, **NECESITAS DEDICAR UNA PORCIÓN DE TIEMPO PARA REPASAR CADA EXAMEN**.

COGE UNA PEGATINA PARA CADA PORCIÓN DE TIEMPO. Usa el botón de cada color para coger (añadir) las pegatinas. Si te equivocas pulsa en Reiniciar, y cuando tengas todas las necesarias pulsa en Enviar.

Enviar Reiniciar

Ciencias naturales
Ciencias Sociales
Matemáticas
Lengua
Inglés

Figura 9. Enunciado y pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia

La interacción que se realiza con la agenda y el formato de respuesta a través de pegatinas se puede ver en la figura 10.

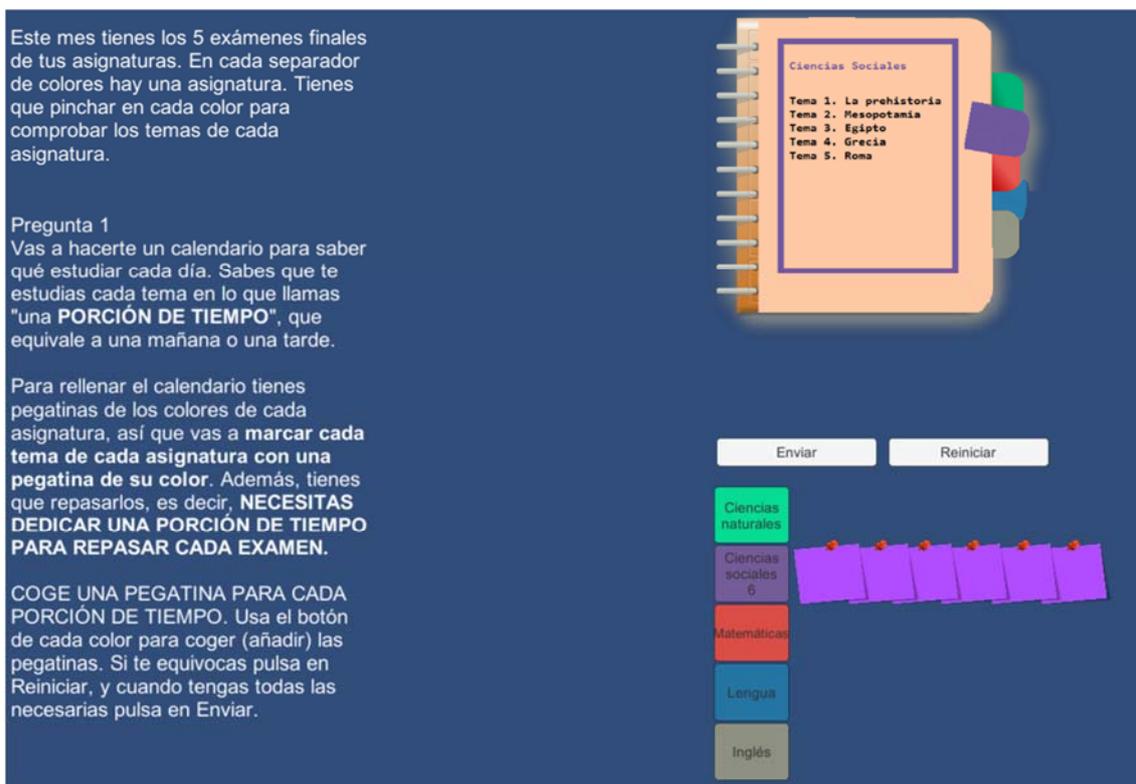


Figura 10. Interacción con la agenda y vista de las pegatinas de la pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia

Esta pregunta tiene tres puntuaciones posibles. La puntuación máxima, 2 puntos, se consigue si cogen una pegatina por cada tema de la asignatura correspondiente – la que tenga ese mismo color – Y además cogen una pegatina para repasar el temario del examen. La puntuación intermedia, 1 punto, se consigue si cogen una pegatina por cada tema de la asignatura correspondiente pero no cogen pegatina para repasar el temario de examen. La puntuación mínima, 0 puntos, se da con cualquier otra respuesta.

Pregunta 2

Esta pregunta es de planificar y ejecutar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 2, ya que tienen que planificar todo el mes con una única restricción: la fecha de cada examen, y ejecutar un solo paso a la vez para lograr un

subobjetivo (organizar el estudio cada examen). El enunciado de esta pregunta se puede ver en la figura 11.

Esta pregunta tiene dos posibles puntuaciones. El código 1 se da cuando todas las pegatinas de la asignatura están incluidas en el calendario antes del examen de esa asignatura. El código 0 se da con cualquier otra planificación.

Figura 11. Enunciado de la pregunta 2 de la prueba Agenda basada en el marco MicroFIN. Fuente: elaboración propia.

Prueba VÍDEO

Esta prueba pertenece al enfoque MicroFIN, y tiene un contexto no tecnológico y social. El enunciado de la prueba se puede ver en la figura 12.

Figura 12. Enunciado y pregunta 1 de la prueba Vídeo con enfoque MicroFIN. Fuente: elaboración propia.

Esta prueba consiste en la planificación de un proyecto de vídeo para la asignatura de historia y tiene doble objetivo. Por una parte, el objetivo es que exploren las tareas que tienen que realizar para cada día para un proyecto de la clase de historia en el que tienen que grabar un vídeo, de forma que sepan el tiempo que pueden que dedicar a cada tarea (explorar y comprender). Por otra parte, con esta prueba se evalúa la capacidad de identificar errores en la planificación y saber cuál es el motivo del error para poder ajustar la planificación (observar y reflexionar).

Pregunta 1.

Esta pregunta es de explorar y comprender, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 3, ya que tienen que tener en cuenta una restricción: dedicar el mayor tiempo a cada tarea, y para ello tienen que empezar al terminar las tareas anteriores. El enunciado de esta pregunta se puede ver en la figura 13.

En grupos de 4 tenéis que crear un vídeo para el proyecto de historia, explicando cómo era la vida en la Edad Media. La fecha de entrega es exactamente dentro de 2 semanas (14 días). Para completar el vídeo habéis hecho una lista con todas las tareas que tenéis que hacer, y habéis creado una tabla para visualizar mejor el trabajo. Cada estrella representa una tarea (pincha sobre cada estrella para verla) y cada estrella está colocada en la fecha que debéis acabarla para que os dé tiempo a hacer las siguientes tareas.

1 ★	2	3	4	5 ★★	6	7
8	9 ★★	10 ★	11 ★★	12	13 ★★	14

Escena explicando la situación política.

Pregunta 1

Como hay poco tiempo para hacer el trabajo, tenéis que dedicar el mayor tiempo posible a cada tarea, y para ello, todos tendréis que estar siempre ocupados, es decir, **tendréis que empezar a hacer cada tarea desde que acabéis la tarea anterior.** Entonces, ¿Qué día podremos empezar con la grabación de las escenas?

Selección Día

- ✓ Selección Día
- Día 1
- Día 2
- Día 3
- Día 4
- Día 5
- Día 6

Figura 13. Interacción con el calendario de la pregunta 1 de la prueba Vídeo, basada en el enfoque MicroFIN.
Fuente: elaboración propia

Esta pregunta cuenta con un calendario para consultar la fecha de entrega de tareas, que se muestran al pinchar sobre cada estrella. Cada estrella corresponde a una tarea. Para la respuesta, cuenta con un desplegable del 1 al 14 para que los alumnos

seleccionen de dicho desplegable la solución correcta. La interacción con la prueba se puede ver en la figura 5.

Esta pregunta tiene dos puntuaciones posibles. El código 1, se da cuando responde que el día que empiezan con las grabaciones es cuando terminan la tarea anterior. Por tanto, se dará por válido que respondan tanto el día 5 como el día 6. El código 0, se da con cualquier otra respuesta.

Pregunta 2.

Esta pregunta es de observar y reflexionar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 2 y 3, según la puntuación obtenida en la respuesta. El enunciado de esta pregunta y la información que se proporciona (la propuesta de Pablo) se puede ver en la figura 14.

En el nivel 3, son capaces de monitorizar el resultado tratando con una condición, la condición de quién hace cada tarea en qué momento. En este nivel serán capaces de detectar que durante los días 6, 7 y 8; Lucía está en las dos tareas y Pablo en ninguna.

En grupos de 4 tenéis que crear un vídeo para el proyecto de historia, explicando cómo era la vida en la Edad Media. La fecha de entrega es exactamente dentro de 2 semanas (14 días).
Para completar el vídeo habéis hecho una lista con todas las tareas que tenéis que hacer, y habéis creado una tabla para visualizar mejor el trabajo. Cada estrella representa una tarea (pincha sobre cada estrella para verla) y cada estrella está colocada en la fecha que debéis acabarla para que os dé tiempo a hacer las siguientes tareas.

1 ★	2	3	4	5 ★★	6	7
8	9 ★★	10 ★	11 ★★	12	13 ★★	14

Pregunta 2

Os ha dicho el profesor que tenéis que hacer todas las tareas en grupo, aunque sea por parejas, pero ninguna tarea puede hacerse de forma individual. Pablo ha propuesto dividir las tareas entre los 4 como aparece en la tabla de la derecha. ¿Crees que hay algún error en el reparto de tareas **entre los miembros del grupo**? Di brevemente dónde está el error.

Escribe aquí tu respuesta...

Tarea	Día en que la tarea tiene que estar terminada	Días en los que hace la tarea	Quién hace cada tarea
Decidir qué queremos contar y cómo	1	1	Todos
Buscar información sobre la Edad Media	5	2-5	Pablo y tú
Crear una introducción al vídeo	5	2-5	Lucía y Adriana
Escena explicando la situación política	9	6-8	Lucía y tú
Escena explicando costumbres y herramientas	9	6-8	Lucía y Adriana
Escena disfrazados de gente de la época	10	9-10	Todos
Edición: cortar partes malas y tomas falsas	11	11	Pablo y tú
Ordenar escenas: secuenciar el trabajo	11	11	Lucía y Adriana
Audio: poner música de fondo	13	13	Pablo y tú
Narrar el vídeo dando sentido a las escenas	13	13	Lucía y Adriana

Figura 14. Enunciado de la pregunta 2 de la prueba Vídeo basada en el enfoque MicroFIN. Fuente: elaboración propia

En el nivel 2, tienen cierta capacidad para monitorizar el resultado, pero sólo logran entender una pequeña parte de todo el escenario del problema. En este nivel serán capaces de detectar que Lucía trabaja más que el resto o que Pablo trabaja menos que el resto, pero sin tener en cuenta la simultaneidad de tareas de Lucía los días 6, 7 y 8.

Por tanto, esta pregunta tiene 3 posibles puntuaciones. La puntuación máxima, es decir, 2 puntos, que corresponde al nivel 3 en el que identifican dónde está el fallo de la planificación.

La puntuación intermedia, es decir, 1 punto, que corresponde al nivel 2 en el que intuyen el problema, pero no saben exactamente dónde se encuentra el fallo de la planificación. Por último, la puntuación 0 se da con cualquier otra respuesta.

Prueba BILLETES

Esta prueba pertenece al enfoque MicroDYN, y tiene un contexto tecnológico y personal. El enunciado de la prueba se puede ver en la figura 15. Para un mayor conocimiento sobre esta prueba, ver las preguntas liberadas en Instituto Nacional de Evaluación Educativa (2013).

BILLETES

Una estación de tren tiene una máquina automática de venta de billetes. Usa la pantalla táctil de la derecha para comprar un billete. Debes elegir tres cosas.

- o Elige la red de trenes que quieres (metro o regional).
- o Elige el tipo de tarifa (normal o reducida).
- o Elige un billete para un día o un billete para un número específico de viajes. Los billetes para un día permiten viajes ilimitados durante el día de su adquisición. Si compras un billete con un número específico de viajes, puedes usar los viajes en diferentes días.

El botón COMPRAR aparece cuando has hecho estas tres elecciones. Hay un botón de CANCELAR que puede usarse en cualquier momento ANTES de presionar el botón COMPRAR.

Selecciona la red de trenes

METRO URBANO TRENES REGIONALES

CANCELAR

TRENESZED

Figura 15. Enunciado de la prueba Billetes de PISA. Fuente INEE (2013)

Pregunta 1.

Esta pregunta evalúa la fase de planear y ejecutar. Según los análisis realizados por el programa PISA, esta pregunta se situaría en el nivel 3. El enunciado de esta pregunta se puede ver en la figura 16.

Esta pregunta tiene dos puntuaciones posibles: 1 si han comprado el billete adecuado, y 0 si no han comprado el billete adecuado. Para que el billete sea el adecuado, es necesario que compren un billete para un tren regional, con tarifa normal, y con dos viajes individuales.



Figura 16. Enunciado de la pregunta 1 de la prueba Billetes de PISA. Fuente: INEE (2013)

Pregunta 2.

Esta pregunta evalúa la fase de explorar y comprender. Según los análisis realizados por el programa PISA, esta pregunta se situaría en los niveles 2 y 5, en función de la puntuación obtenida. El enunciado de esta pregunta se puede ver en la figura 17.

Esta pregunta tiene tres posibles puntuaciones. La puntuación máxima (2 puntos) se otorga cuando compran un billete para el metro, con una tarifa reducida y cuatro viajes individuales, y además han comparado el precio con el billete para un día en la misma tarifa, que se supone si visitan la pantalla para comprar este billete. La puntuación intermedia (1 punto) se da cuando compra cualquier billete para metro con tarifa reducida, pudiendo ser individual o diario. Pero en el caso de ser individual, no ha comparado el precio de ambos billetes, ya que si no obtendría la puntuación máxima. Para cualquier otra respuesta, la puntuación es 0.

Pregunta 2: BILLETES CP038Q01

Hoy piensas hacer cuatro viajes por la ciudad en metro. Eres estudiante, así que puedes usar las tarifas reducidas.

Usa la máquina de venta de billetes para encontrar el billete más barato y pincha en COMPRAR.

Una vez que hayas pinchado en COMPRAR, no podrás volver a la pregunta.



Figura 17. Enunciado de la pregunta 2 de la prueba Billetes de PISA. Fuente: INEE (2013)

Pregunta 3.

Esta pregunta evalúa la fase de observar y reflexionar. Según los análisis realizados por el programa PISA, esta pregunta se situaría en el nivel 4. El enunciado de esta pregunta se puede ver en la figura 18.

Pregunta 3: BILLETES CP038Q03

Quieres comprar un billete de metro urbano con dos viajes individuales. Eres estudiante, así que puedes usar tantas reducidas.

Usa la máquina de venta de billetes para adquirir el mejor billete disponible.



Figura 18. Enunciado de la pregunta 3 de la prueba Billetes de PISA. Fuente: INEE (2013)

En esta pregunta se encuentran con el problema de que el billete que tienen que intentar comprar no está disponible. En este caso, deberán explorar otras opciones de billetes para el metro. El billete con tarifa reducida para viajes ilimitados en el día (diario) y las opciones de billete con tarifa normal tanto para 2 viajes individuales como el billete diario. Después de comparar, deberán comprar el mejor billete.

Esta pregunta tiene dos puntuaciones: 1 punto si compras el billete disponible más adecuado, que por el precio es el billete con tarifa normal para 2 viajes individuales; y 0 puntos si compras cualquier otro billete.

Prueba TREN

Esta prueba pertenece al enfoque MicroFIN, y tiene un contexto no tecnológico y social. El enunciado de la prueba, junto a la pregunta 1, se pueden ver en la figura 19.

Esta prueba consiste en organizar un viaje en tren por Europa. Por una parte, el objetivo de la prueba es averiguar el recorrido a partir de unas anotaciones incompletas que muestran el tiempo de viaje entre ciudades (representar y formular). Por otra parte, el objetivo de la prueba consiste en identificar dos fallos cometidos en la planificación de los hoteles para el viaje (observar y reflexionar).

Pregunta 1.

Esta pregunta es de representar y formular, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 4, ya que tienen que representar el recorrido del viaje probando sistemáticamente diferentes posibilidades y chequeando si se cumplen varias condiciones simultáneas, al explorar los diferentes tiempos de viajes de cada ciudad. El enunciado de esta pregunta se puede ver en la figura 11.

¡Nos vamos de viaje! Este verano vamos a hacer un interrail por Europa, viajando en tren por un montón de países.
El mapa de la derecha muestra las horas que se tardan viajando entre los destinos señalados.

Pregunta 1. Ya hemos mirado los días que queremos estar en cada ciudad, y las horas de los trenes para movernos entre las ciudades. Pero por un pequeño problema al guardar el archivo, nos falta la parte en la que verifican las ciudades, por lo que no sabemos el recorrido. Reconstruye el recorrido del viaje mirando el mapa a partir de las horas de viaje.

Nota: Sales desde Barcelona.

Días en la ciudad (Aproximados)	Hora salida del tren	Horas de viaje
3	9:20	4h 25min
5	12:55	1h 45min
4	9:21	2h 20min
4	7:01	6h 05min
5	8:46	4h 40min
3	8:42	4h 10min
4	19:17	15h 55min
3	Avión a Madrid: miércoles 31. Hora 21:20	

Barcelona ▾ Elige ciudad ▾

Enviar ▾
Ljubliana
Londres
Madrid
Milán
Niza
París
Praga

Figura 19. Enunciado de la prueba Tren y de la pregunta 1, prueba basada en el marco MicroFIN. Fuente: elaboración propia.

Esta pregunta tiene dos posibles puntuaciones. La puntuación 1 se da si se ha representado adecuadamente el recorrido, por lo que todas las ciudades deben estar en orden. La puntuación 0 se da con cualquier otra representación.

Pregunta 2.

Esta pregunta es de observar y reflexionar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 5, ya que en este nivel los alumnos pueden detectar errores o dificultades inesperadas para ajustar sus planes. Tanto la puntuación máxima como la intermedia se sitúan en el nivel 5, ya que en ambos casos implica detectar un error. El enunciado de esta pregunta se puede ver en la figura 20.

[Nos vamos de viaje! Este verano vamos a hacer un interrail por Europa, viajando en tren por un montón de países. La tabla de la derecha muestra las horas de salida más las horas de viaje, para saber la hora a la que llegamos al siguiente hotel.

Pregunta 2. Hemos hecho las reservas del hotel. Pero no estamos seguros si lo hemos reservado correctamente porque no nos hemos fijado en la hora de salida de los trenes.

Compara las fechas de los hoteles de la tabla de reservas con la tabla donde hemos apuntado los horarios de los trenes. MARCA EN LA TABLA SI HEMOS RESERVADO UNA NOCHE DE HOTEL QUE NO NECESITAMOS. Y/O SI NOS HA FALTADO POR RESERVAR ALGUNA NOCHE DE HOTEL, y cuando lo tengas dale a Enviar. Para marcar, pulsa sobre la fecha de reserva incorrecta y aparecerá un asterisco (*).

NOTA: 1 día en el hotel significa que tenemos la habitación desde las 13.00 horas de ese día hasta las 12.00 horas del día siguiente, y podemos entrar después y salir antes, respectivamente.

Dias en la ciudad (Aproximados)	Hora salida del tren	Horas de viaje
3	9.20	6h 25min
5	12.55	1h 43min
4	9.21	2h 20min
4	7.01	6h 05min
5	8.46	4h 40min
3	8.42	4h 10min
4	19:17	15h 55 min
3	Avión a Madrid: miércoles 31. Hora 21.20	

Ciudad	Dia de Entrada	Dia de salida
Barcelona (España)	1	4
París (Francia)	4	8 *
Bruselas (Bélgica)	9	13
Ámsterdam (Holanda)	13	17
Berlín (Alemania)	17	22
Praga (Rep. Checa)	22	25
Viena (Austria)	25	29 *
Milán (Italia)	29	31

Figura 20. Enunciado de la pregunta 2 de la prueba Tren basada en el marco MicroFIN. Fuente: elaboración propia.

Esta pregunta tiene tres posibles puntuaciones. La puntuación máxima, 2 puntos, se da cuando detectas el error de que en París te quedas el último día sin hotel (la noche del 8) y también detectas el error de que en Milán no necesitas el primer día de hotel porque pasas esa noche en el tren desde Viena (la noche del 29). La puntuación

intermedia, 1 punto, se da cuando detectas un error u otro. La puntuación inferior, 0, se da si no detectas ninguno de los dos errores.

Para responder basta con marcar el día de salida de un hotel (el día 8 y el día 29) o el día de entrada al siguiente hotel (el día 9 y el día 29).

Prueba CLIMATIZADOR

Esta prueba pertenece al enfoque MicroDYN, y tiene un contexto tecnológico y personal. El enunciado de la prueba se puede ver en la figura 21. Para un mayor conocimiento sobre esta prueba, ver las preguntas liberadas en Instituto Nacional de Evaluación Educativa (2013).

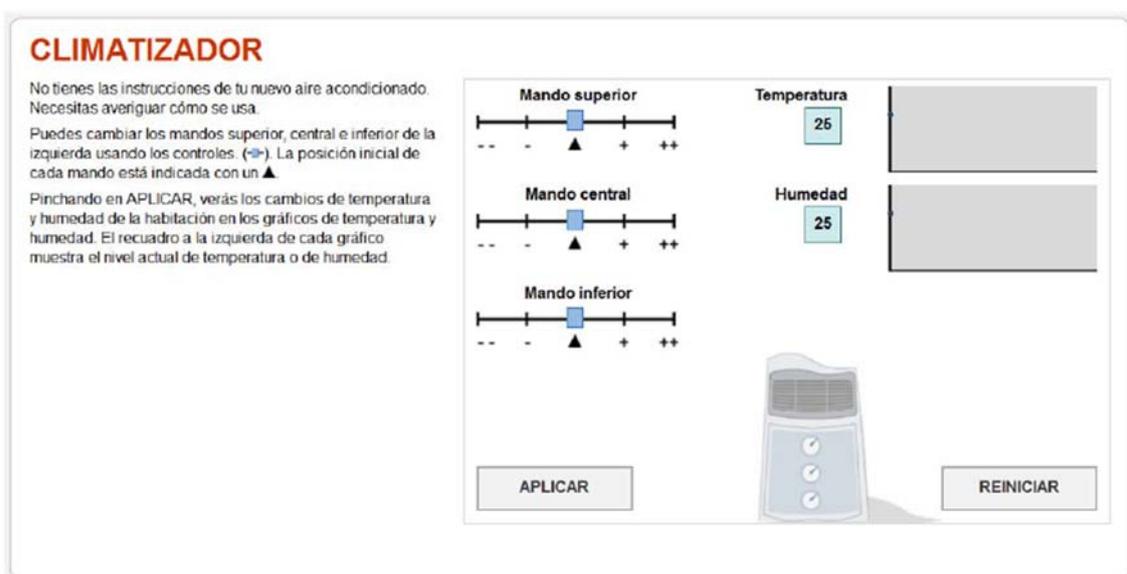


Figura 21. Enunciado de la prueba climatizador de PISA. Fuente: INEE (2013)

Pregunta 1.

La primera pregunta de Climatizador es sobre representar y formular, y según los análisis realizados por el programa PISA, esta pregunta se situaría en el nivel 3. El enunciado de esta pregunta se puede ver en la figura 22.

Pregunta 1: CLIMATIZADOR CP025Q01

Averigua si cada mando influye sobre la temperatura y humedad al variar los controles. Puedes empezar otra vez pinchando en REINICIAR.

Dibuja líneas en el diagrama de la derecha para indicar en qué influye cada mando.

Para dibujar una línea, pincha sobre un mando y luego pincha en Temperatura o en Humedad. Puedes eliminar cualquier línea pinchando sobre ella.

Figura 22. Enunciado de la pregunta 1 de la prueba climatizador de PISA. Fuente: INEE (2013)

Para el programa PISA, esta pregunta tenía 3 posibles puntuaciones, ya que la puntuación intermedia se otorga al haber utilizado una estrategia específica para tratar la información (vary-one-thing-at-time o VOTAT).

En nuestro caso, esta pregunta sólo tiene dos posibles puntuaciones (código 1 si es correcta, código 0 si no es correcta). Para que la pregunta se dé por correcta, la persona tiene que construir el modelo correcto, relacionando el mando superior con la temperatura, y los mandos central e inferior con la humedad. Si el modelo no es correcto, obtendrá el código 0.

Pregunta 2.

La segunda pregunta de Climatizador es sobre planear y ejecutar, y según los análisis realizados por el programa PISA, esta pregunta se situaría en los niveles 4 y 5. El enunciado de esta pregunta se puede ver en la figura 23.

Pregunta 2: CLIMATIZADOR CP025Q02

A la derecha se muestra la relación correcta entre los tres mandos, la Temperatura y la Humedad.

Usa los mandos para regular la temperatura y la humedad hasta los niveles deseados. **Hazlo en cuatro pasos como máximo.** Los niveles deseados se muestran mediante bandas rojas en los gráficos de Temperatura y Humedad. El intervalo de los valores para cada nivel deseado es 18-20 y se muestra a la izquierda de cada banda roja. **Sólo puedes pinchar en APLICAR cuatro veces y no hay botón de REINICIAR.**

Figura 23. Enunciado de la pregunta 2 de la prueba climatizador de PISA. Fuente: INEE (2013)

Esta pregunta tiene 3 posibles puntuaciones. La máxima puntuación, un 2, se da cuando el resultado de ambas variables está entre los valores 18 y 20. La puntuación intermedia, un 1, se da cuando el resultado no está entre los valores 18 y 20, pero la distancia a estos valores es menor de 5 en el resultado de ambas variables. Cualquier otra respuesta tiene la puntuación 0.

Prueba COCHE

Esta prueba pertenece al enfoque MicroDYN, y tiene un contexto tecnológico y personal. El enunciado de la prueba se puede ver en la figura 24.

Se trata de un sistema complejo con múltiples relaciones entre los distintos elementos del problema. Este sistema representa un coche híbrido en el que puedes elegir en todo momento el modo de conducción que deseas utilizar (híbrido, gasolina, eléctrico).

La prueba consiste en comprender la interacción entre las variables de entrada y las variables de salida del coche para llegar a una fiesta. Las variables de entrada son los modos de conducción: híbrido, gasolina y eléctrico. Las variables de salida son el progreso de avance y la energía.

El progreso de avance tiene que completarse, ya que indica que has llegado al destino, por lo que es nuestro objetivo; como cada modo de conducción avanza de modo distinto, tienes que tener en cuenta cuánto avanza cada modo para poder llegar.

Por otra parte, la energía es la que permite que puedas avanzar, por lo que digamos que es una restricción: para llegar al final tienes que tener energía, por lo que tienes que tener en cuenta cuánta energía gasta cada modo de conducción.

Además de controlar el efecto de las variables de entrada en ambas variables de salida, las variables de entrada se influyen entre ellas. Es decir, al utilizar el modo híbrido también disminuye tu modo gasolina. Al utilizar el modo gasolina, afecta también al

modo eléctrico. Por último, al utilizar el modo eléctrico, influye en la variable de modo híbrido pero de una forma positiva, es decir, aumenta el modo híbrido.

Esta prueba tiene múltiples objetivos: descubrir la relación entre las variables (explorar y comprender), representar la relación (representar y formular), llegar a la fiesta – completando la barra de progreso de avance – (planear y ejecutar), e identificar cuántas veces podremos usar el modo Gasolina (observar y reflexionar).

La relación entre las variables se rige por las siguientes reglas:

1. Cada modo de conducción resta el 20% de su propia barra de progreso. Por tanto, si lo calculamos sobre 10 posiciones o unidades, utilizar un modo supone quitarle 2 unidades.
2. El progreso de avance y la energía tienen 20 posiciones o unidades. El progreso aumenta, y la energía disminuye, a diferente ritmo según el modo de conducción elegido.
3. Cada uno de los modos, influye en las demás variables en lo siguiente:
 - Híbrido: gasolina (-1 ud.) / energía (-3 uds.) / avanzar (+3 uds.)
 - Gasolina: eléctrico (-1 ud.) / energía (-2 uds.) / avanzar (+2 uds.)
 - Eléctrico: híbrido (+1 ud.) / energía (-1 ud.) / avanzar (+1 ud.)

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Progreso de avance

Energía

Gasolina Eléctrico Híbrido

Reiniciar

Figura 24. Enunciado de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

Pregunta 1.

Esta pregunta es de explorar y comprender, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 4 y 6, ya que la puntuación parcial equivale a comprender las relaciones directas con las variables de resultado, temperatura y humedad, lo que correspondería a un nivel 4; y la puntuación máxima equivale a comprender además las relaciones indirectas con otras las variables que afectan a la temperatura y la humedad, lo que se corresponde con un nivel 6. El enunciado de esta pregunta se puede ver en la figura 25.

Esta pregunta tiene tres puntuaciones posibles. La puntuación máxima, es decir, 2 puntos, se da cuando marcan todas las relaciones: relaciones directas (energía y progreso), y relaciones indirectas (híbrido). La puntuación intermedia, es decir, 1 punto, se da cuando marcan las relaciones directas: energía y progreso. La puntuación mínima, 0 puntos, se da con cualquier otra respuesta.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Pregunta 1
Averigua a cuáles de los elementos del problema afecta conducir el coche en modo eléctrico.

Energía
 Progreso de avance
 Gasolina
 Híbrido

Enviar

Progreso de avance

Energía

Gasolina Eléctrico Híbrido

Reiniciar

Figura 25. Enunciado de la pregunta 1 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

Pregunta 2.

Esta pregunta es de representar y formular, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 5 y 6, ya que la puntuación intermedia implica explorar un problema para comprender cómo se estructura la información relevante, en este caso, las relaciones directas; y la puntuación máxima implica comprender cómo se estructura toda la información del problema. El enunciado de esta pregunta aparece en la figura 26.

Esta pregunta tiene tres posibles puntuaciones. La puntuación total, 2 puntos, se consigue si representan todas las relaciones directas: energía y progreso con cada modo de conducción, y las relaciones indirectas: entre gasolina, híbrido, eléctrico; es decir, se consigue si representan adecuadamente el modelo que aparece en la figura 27.

La puntuación intermedia, 1 punto, se consigue si representan sólo las relaciones directas, es decir, los 3 modos de conducción con el objetivo -progreso de avance- y con la restricción -energía-). La puntuación 0 se da con cualquier otro modelo representado.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Pregunta 2

Averigua a qué afecta cada modo de conducción. Dibuja líneas en el diagrama de abajo para indicar en qué influye cada modo de conducción.

Para dibujar una línea, pincha sobre un modo de conducción y luego pincha en Energía, Progreso de avance u otro modo de conducción. Puedes eliminar cualquier línea pinchando sobre ella.

The diagram shows two columns: 'Energía' and 'Progreso de avance'. Below these are three rows for 'Gasolina', 'Eléctrico', and 'Híbrido'. Each row has a bar with a black segment on the left and an orange segment on the right. A 'Reiniciar' button is below the diagram. Below the diagram is another diagram with three boxes for 'Gasolina', 'Eléctrico', and 'Híbrido' on the left, and two boxes for 'Energía' and 'Progreso de avance' on the right. A black arrow points from 'Gasolina' to 'Energía'. An 'Enviar' button is at the bottom.

Figura 26. Enunciado de la pregunta 2 de la prueba Coche basada en el marco MicroDYN. En negro aparecen las relaciones correctas y en rojo las otras posibles relaciones. Fuente: elaboración propia.

Pregunta 3.

Esta pregunta es de planificar y ejecutar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 6, ya que en este nivel son capaces de crear planes complejos, flexibles y con múltiples pasos, y llevarlos a cabo monitorizando todo el proceso por si tienen que reajustar su plan. El enunciado de esta pregunta está en la figura 27.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.
NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Energía Progreso de avance

Gasolina Eléctrico Híbrido

Pregunta 3

Intenta llegar a la fiesta. Para ello, tendrás que completar la barra de progreso de avance sin quedarte sin energía.
Si no has podido terminar, pulsa enviar.

Gasolina

Eléctrico

Híbrido

Energía

Progreso de avance

Enviar

Figura 27. Enunciado de la pregunta 3 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

El nivel 6 concretamente correspondería con crear un plan eficaz, que permita llegar con el mínimo número de pasos. También pueden completar el progreso de avance sin tener el plan más eficaz, este caso correspondería al nivel 4.

Esta pregunta tiene tres posibles puntuaciones. La puntuación máxima, correspondiente a 2 puntos, se da cuando consigue llegar a la fiesta, es decir, que la barra de progreso de avance se complete, en el menor número de pasos posible, que es 8 pasos (2 pasos en modo eléctrico y 6 pasos en modo híbrido). La puntuación intermedia, correspondiente a 1 punto, se da cuando consigue llegar a la fiesta, siendo indiferente el número de pasos que haya necesitado para llegar. La puntuación mínima, correspondiente

a 0 puntos, se da cuando no se llega a la fiesta porque la barra de energía se acaba antes de completar la barra de progreso.

Pregunta 4.

Esta pregunta es de observar y reflexionar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 6, ya que en este nivel son capaces de desarrollar modelos mentales con diferentes escenarios del problema y simular la ejecución de un plan para reconocer de forma anticipada si es posible lograr el objetivo o no. El enunciado de esta pregunta se muestra en la figura 28.

Esta pregunta tiene dos posibles respuestas. El código o puntuación 1, se da cuando señalan la opción 3 veces, ya que para llegar a la fiesta solo pueden usar 3 veces como máximo el modo de conducción Gasolina. El código o puntuación 0 se da con cualquier otra respuesta.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Pregunta 4

Mi tío me ha dicho que seguramente pueda llegar en el modo Gasolina, pero yo no lo creo. Observa de nuevo cómo interacciona el sistema de modos de conducción y reflexiona sobre cuántas veces puedes utilizar **como máximo** el modo Gasolina.

0
 1
 2
 3
 4
 5

Enviar

Energía Progreso de avance

Gasolina Eléctrico Híbrido

Reiniciar

Figura 28. Enunciado de la pregunta 4 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

Curso de pensamiento computacional

Se ha realizado un curso de pensamiento computacional que nos ha permitido el estudio de la variable independiente, de forma que podemos afirmar que, quienes han asistido a este curso han desarrollado el pensamiento computacional, ya que se ha creado en base a las investigaciones sobre didáctica del pensamiento computacional de Marais y Bradshaw (2015).

Este curso de pensamiento computacional tiene una duración de 30 horas, y ha sido incluido en una extraescolar de programación que ya se estaba impartiendo y que llevaba 10 horas cuando empezó el curso de pensamiento computacional. Estas 30 horas de formación nos permiten diferenciar dos grupos en la variable independiente.

Para el diseño del curso de pensamiento computacional, se han tenido en cuenta dos aspectos:

Por una parte, debemos enseñar las estrategias que llevan a cabo los científicos de la computación cuando se encuentran ante un problema de su campo. Estas estrategias, aplicadas al aprendizaje de la programación, son:

1. **Automatización.** Para ello, tendrán que incluir instrucciones de control en su código, como condicionales y bucles, en lugar de algoritmos simples.
2. **Descomposición de los problemas.** En este punto se insiste en que no deben abordar el problema como un todo, si no que deben intentar identificar las diferentes partes del problema para abordarlas de forma independiente.
3. **Simulación.** Para desarrollar habilidades en simulación, tienen que ser capaces de proponer una solución y simular los pasos que harían hasta alcanzar la solución, de forma que puedan anticipar cuando una solución no va a ser válida.

4. **Paralelismo.** Debido a las características de los problemas complejos hace necesario trabajar en varias capas de información. A partir de este supuesto, el paralelismo se debe enfocar hacia la simulación y la descomposición de los problemas. Esto que decir que los alumnos deberán ser capaces de simular las soluciones a diferentes partes del problema a la vez y, en especial, deberán ser capaces de engranar todos esos procesos en su simulación para acercarse lo más posible a la solución del problema.

Por otra parte, debemos tener en cuenta dónde se aplican estas habilidades, y aquí nos encontramos con la capacidad de resolución de problemas. Para trabajar la aplicación de este pensamiento, se aplican las estrategias que conlleva dicho pensamiento a través de unos pasos clave, que se harán explícitos cada vez que les presente un reto o piensen en un proyecto.

Los pasos clave que se han utilizado para el curso se han cogido de la propuesta que hacen Marais y Bradshaw (2015) y se describen a continuación:

1. **Lograr un claro estado de ánimo.** Este paso se refiere a que no entren en pánico cuando vean un problema. Que piensen que van a poder resolverlo, que toda la información que necesitan va a estar disponible o accesible fácilmente, y que tienen que creer en ellos mismos, en que conocen la estrategia para abordarlo.
2. **Tratar de entender el problema a fondo.** En este paso es cuando se les hace ver la importancia de leer, y releer el problema tantas veces como haga falta, hasta que estén seguros de que saben cuál es el problema y qué tienen que hacer. Tienes que preguntarles qué han leído, qué les estás pidiendo, para ver que su representación del problema es correcta.
3. **Examinar críticamente el problema.** Cuando tienen claro lo que les pide el problema, tienen que explorar el entorno (el enunciado del problema, los recursos físicos disponibles, en el caso de mis pruebas todo lo que era interactivo, etc.) para

comprobar si tienen todos los datos. Con estos datos, tienen que idear una solución, construir un "roadmap" (mapa del problema) para intentar solucionar el problema. Este paso termina con un modelo mental que permite saber lo que tiene que hacer.

4. **Simplificar el problema en base a una solución prevista.** En este paso, se insiste en dos de las estrategias utilizadas por los científicos de la computación: descomposición del problema y automatización.
5. **Identificar una posible ruta hacia la solución.** Este paso de la resolución del problema depende en gran medida de la creatividad de cada persona.
6. **Asegurarse de que el problema puede resolverse.** Por último, se insiste en las otras dos estrategias utilizadas por científicos de la computación: simulación y paralelismo.

Procedimiento

Debido a la naturaleza de la investigación, la medición empleada en este estudio se ha realizado fundamentalmente en dos momentos, realizándose el pretest el día 20 de febrero de 2017 y el posttest el día 25 de abril de 2017.

En un primer momento se procedió a la aplicación de las pruebas de resolución de problemas a todos los alumnos de 4º de Educación Secundaria el lunes 20 de febrero, excepto a cuatro alumnos a los que se les aplicaron las pruebas el siguiente lunes, 27 de febrero.

En estos momentos, los alumnos que participan en la extraescolar de programación, y que nos permitirían diferenciar los grupos de la variable independiente, llevaban 10 horas de formación ya que esta actividad comenzó en enero y se desarrolla los lunes durante 2 horas.

En estas sesiones habían estado explorando de forma libre cómo crear sus proyectos de programación y habían aprendido a crear algoritmos sencillos de

programación, pero sin tener una referencia que les explicase las estrategias que utilizan habitualmente los científicos de la computación para organizar el código.

Por tanto, consideramos que, dadas las horas de formación y el método de enseñanza empleado, exploración libre, no era suficiente como para que se estableciesen diferencias entre ambos grupos. Aun así, para descartar estas diferencias, se van a interpretar los datos de la fase pretest, de cara a confirmar nuestras suposiciones.

El mismo lunes 20 de febrero comenzó el curso de pensamiento computacional, aprovechando que los lunes son los días en los que está programada la actividad extraescolar. Esta sesión fue impartida por la investigadora, de forma que el profesor responsable de la actividad pudiese observar las pautas clave que se les daba a los asistentes a la actividad.

Una vez finalizada la primera sesión, se impartió igualmente una formación de dos horas al profesor responsable de la actividad, repasando los puntos en los que debía instruir a los participantes de cara al desarrollo del pensamiento computacional.

Además, en esta formación se comentaron posibles casos y se resolvieron dudas. Igualmente, se ofreció la posibilidad de contacto al profesor tanto por correo electrónico como por teléfono móvil, de forma que pudiese resolver sus dudas en el momento en el que surgen.

Las siguientes sesiones fueron todos los lunes desde el 20 de febrero hasta el 24 de abril, más una sesión extra el jueves 6 de abril, justo antes de las vacaciones de semana, por petición de los propios alumnos, ya que la finalidad de la actividad extraescolar era presentarse a un concurso tecnológico en el que compiten con proyectos electrónicos similares.

El segundo momento de aplicación de las pruebas, el posttest, se realizó el 25 de abril a los 39 participantes, dando por terminada la formación en pensamiento computacional para dicho momento.

En el momento que se aplicaron las pruebas por segunda vez, se procede al análisis de datos para comprobar las hipótesis planteadas para la investigación, cuyos resultados se recogen en el capítulo de resultados.

Capítulo 6. Validación de las pruebas de RPC

Para entender los motivos que respaldan la elección de un marco concreto de cara a la creación de las pruebas, de forma que éstas conformen un instrumento sólido de evaluación de la habilidad computacional en la resolución de problemas complejos, en este capítulo vamos a hacer en primer lugar un breve repaso y recopilar todos los requisitos y características que nos han llevado a su elección.

La resolución de problemas complejos (RPC, por las siglas de su denominación en inglés *Complex Problem Solving*), es una parte de la resolución de problemas, cuyo enfoque de investigación se ha ido distanciando de la investigación de resolución de problemas tradicional, debido a las características propias que contienen este tipo de problemas denominados complejos.

Funke y Frensch (1995) proponen una definición de los problemas complejos que suponen a la persona que lo resuelve unos requisitos concretos para poder ser resueltos. Principalmente estas peculiaridades residen en que, para resolver un problema complejo, la persona no dispone de todos los datos del contexto desde el inicio del problema.

Este motivo hace que necesite evaluar continuamente cada estado del problema en el que se encuentra desde que parte del estado inicial, para recoger nuevos datos y valorar si ha tomado un buen camino hacia el estado objetivo o debe reformular el problema debido a que cambia su representación sobre el mismo.

Al no tener los datos desde el inicio del problema, los problemas complejos también se recogen en la literatura como problemas dinámicos, diferenciándose así de los problemas estáticos (Herde, Wüstenberg y Greiff, 2016).

La continua interacción con el problema debe resultar eficiente, o por el contrario no se avanzará hacia la resolución del problema. Y para que sea eficiente, se requiere

detectar todos los datos relevantes del problema y además establecer relaciones lógicas entre los distintos elementos, lo que requiere unas capacidades superiores de resolución de problemas.

Las características particulares que definen a los problemas complejos, requieren unas estrategias más elaboradas que las empleadas tradicionalmente en la resolución de problemas, para enfrentarse a este tipo de problemas de forma que logremos resolverlos. De ello se deduce que estudiar qué estrategias son más adecuadas para abordar problemas complejos es fundamental para poder abordar problemas complejos de forma adecuada.

Y para realizar el estudio de estas estrategias debemos generar unos instrumentos de evaluación diferentes a los utilizados tradicionalmente para el estudio de resolución de problemas, de forma que nos permitan investigar cómo se estructura la resolución de problemas complejos. Así mejorará el conocimiento sobre el empleo de diferentes estrategias, lo que nos ayudará a determinar cuáles son las estrategias más efectivas ante este tipo de problemas.

La resolución de problemas complejos, como tema de investigación independiente, se ha empezado a estudiar con más fuerza durante estos últimos años. Como recoge Funke (2013), la investigación sobre resolución de problemas ha aumentado en gran medida durante los últimos años, y con la llegada de nuevos métodos para el rastreo de procesos, la creación de micromundos virtuales y la definición de nuevos constructos, las teorías están pudiendo ser ajustadas.

Para evaluar la resolución de problemas complejos se han propuesto marcos de evaluación formales que han permitido mejorar la validez de esta evaluación. Por tanto, las pruebas que se validan en este artículo se fundamentan en las tendencias de evaluación de la RPC de estos últimos años.

Inicialmente, Funke (2001) propuso los enfoques originales MicroDYN y MicroFIN, siendo uno de ellos cuantitativo y el otro cualitativo, respectivamente, para la evaluación de la resolución de problemas complejos. El enfoque de las pruebas MicroDYN se basa en el marco de sistemas lineales de ecuaciones estructurales (LSE), mientras que el enfoque de MicroFIN se basa en el marco de sistemas autómatas de estados finitos (FSA).

Más recientemente, Greiff, Wüstenberg y Funke (2012) han propuesto un enfoque de *sistemas complejos múltiples* (MCS en inglés), que aúna ambos marcos, y que permite aumentar su complejidad en cuanto a la evaluación (Herde, Wüstenberg y Greiff, 2016).

Su propuesta consiste en introducir sistemas informáticos que permitan reproducir y medir la interacción entre el problema y la persona que lo resuelve, de una forma más sencilla y fiable que las pruebas de lápiz y papel.

Las tres ventajas principales del enfoque de sistemas complejos múltiples son: (1) permite discriminar varios niveles de competencia, (2) estima con una alta fiabilidad la adquisición del conocimiento y la aplicación del conocimiento, y (3) limita dificultades en las puntuaciones finales de RPC debidas a errores aleatorios aislados al comienzo de la evaluación (Herde, Wüstenberg y Greiff, 2016).

Este enfoque se incorporó parcialmente en las tareas utilizadas para la evaluación de la RPC en el *Programme for International Student Assessment* (Herde, Wüstenberg y Greiff, 2016), programa conocido popularmente como las pruebas PISA, en su edición de 2012, ya que se basó en pruebas digitales en las que los alumnos interactúan de forma continua con los datos de problemas dinámicos.

Este enfoque, el enfoque de los sistemas múltiples complejos, resulta adecuado para el estudio del empleo de estrategias de nivel superior en la resolución de problemas

complejos, ya que este enfoque nos permite distinguir entre una primera fase de adquisición del conocimiento sobre el problema y una segunda fase relacionada con la aplicación del conocimiento para el control del problema (Funke, 2001).

El empleo de estrategias puede darse en ambas fases de la resolución de problemas, o sólo en una de ellas, lo que nos va a permitir profundizar en el conocimiento sobre si una estrategia es adecuada para la adquisición del conocimiento, para la aplicación del mismo, o para ambas.

La creación del instrumento de evaluación que se analiza en este artículo pretende comprobar cómo mejora la resolución de problemas complejos con el uso del pensamiento computacional, queriendo obtener además un conocimiento más profundo de este asunto indagando en si esta estrategia es útil para la fase de adquisición de conocimiento sobre el problema, para la fase de aplicación del conocimiento para el control del problema, o para ambas fases.

Evaluación del PC en la resolución de problemas complejos

El pensamiento computacional (Wing, 2006) en su origen se definió como un pensamiento que influye en la planificación y ejecución de los problemas complejos; al utilizar técnicas que emplean habitualmente los científicos de la computación, facilitando la resolución de los mismos.

En este caso, el pensamiento computacional se emplea como un pensamiento estratégico basado en un enfoque computacional de cara al establecimiento de un plan (Schoppek, 2002).

Posteriormente, surgen evidencias a favor (Kwisthout, 2012; Wing, 2014) de que es necesario un modelo computacional para la representación de problema complejos, ya

que sería más fácil gestionar gran cantidad de variables consideradas de una relevancia similar entre ellas, y se facilitaría así la resolución de problemas.

En este otro caso, el pensamiento computacional se emplea como un modelo de procesamiento de información, una estrategia de pensamiento de nivel superior que es capaz de computar un gran modelo para facilitar su representación.

Por tanto, las investigaciones posteriores sugieren que también se utiliza una estrategia computacional para construir esa representación, por lo que podríamos decir que el pensamiento computacional también facilita la representación de problemas complejos, y no sólo se aplica al proceso de resolución como se había planteado inicialmente.

Por ello, resulta necesario investigar y conocer cómo este pensamiento, o estrategia de pensamiento para ser más exactos, facilita la resolución de problemas complejos, es decir, necesitamos conocer en qué fases de la RPC está implicado el pensamiento computacional, ya que no tendremos una definición del pensamiento computacional adecuada hasta que no conozcamos cómo puede influir éste en cada parte de la resolución de problemas complejos.

Así, nuestra intención es explorar el concepto de pensamiento computacional para averiguar su alcance en la resolución de problemas y establecer una definición adecuada de este pensamiento, ya que, al igual que sugieren Blackwell, Church y Green (2008), es importante conocer las ventajas de adoptar este nuevo estilo de pensamiento en la resolución de problemas, de modo que realmente aporte valor y sepamos aprovechar sus ventajas.

De esta forma, si el instrumento de evaluación que se analiza en este artículo es adecuado para medir la resolución de problemas complejos, podremos estudiar cómo

afecta el pensamiento computacional en la RPC, ya que podremos conocer en qué fases de la resolución de problemas complejos está implicado este pensamiento.

Método para la validación

Diseño de la fase de validación

Se trata de una investigación sobre las propiedades psicométricas de varias pruebas diseñadas para evaluar cómo la estrategia computacional facilita la resolución de problemas complejos. Para ello, se va a medir la consistencia interna con el alfa de Cronbach, de cara a determinar la fiabilidad del instrumento.

Participantes

Los participantes de la investigación son dos muestras de alumnos de 4º ESO, en total 66 participantes. La mayoría de los participantes tiene 15 años, edad para la que ha sido diseñado el marco de PISA 2012, y que se ha tomado de referencia para la creación del instrumento de evaluación, y por tanto, para la batería de las pruebas.

Una de las muestras se ha tomado específicamente para la validación de las pruebas, en la cual hay un total de 28 participantes. La otra muestra corresponde al grupo completo de la investigación principal, es decir, se incluye tanto el grupo experimental como el grupo control, y los datos son los que se han recogido para la fase pre-test de la investigación principal. En esta muestra participan 38 alumnos.

Instrumentos

Marco para el diseño de las pruebas

Como se ha comentado anteriormente, la evaluación de la resolución de problemas ha mejorado gracias a la aplicación del enfoque de sistemas complejos múltiples (MCS). Las pruebas diseñadas para la investigación se han basado en este enfoque de cara a

garantizar una buena validez de constructo. Uno de los marcos que adoptan este enfoque es el propuesto por el programa internacional de evaluación de estudiantes PISA.

El programa PISA mide diferentes aspectos en alumnos de 15 años a nivel internacional, permitiendo realizar comparaciones entre los diferentes países participantes, de cara a detectar puntos débiles en los sistemas de enseñanza y así mejorarlos (Schleicher, 2006).

En la edición de PISA 2012 se incluyen pruebas de resolución de problemas complejos, basadas en el enfoque de sistemas complejos múltiples. Pero no se mantienen en ediciones posteriores. En la edición de PISA 2015, se evalúa la resolución de problemas de forma colaborativa, adoptando un marco diferente al anterior (Greiff, Holt y Funke, 2013).

En nuestro caso, vamos a utilizar el marco de evaluación PISA 2012 (OECD,2010), que se puede ver en la tabla 1, para crear pruebas que conformen nuestro instrumento de evaluación, ya que este marco adopta a las características de sujetos de 15 años, los aspectos de evaluación de los sistemas múltiples complejos (MCS), que supone mejoras respecto a otros enfoques de evaluación (Herde, Wüstenberg y Greiff, 2016).

Por tanto, según las características de los MCS, se han desarrollado pruebas informáticas que reproducen situaciones en las que la persona tiene que interactuar con el problema.

Diseño de las pruebas

El marco utilizado permite la evaluación del uso de estrategias computacionales en la resolución de problemas complejos, ya que las estrategias computacionales, también denominadas pensamiento computacional, se aplicarían directamente en la resolución de problemas, por lo que estas pruebas nos permitirán conocer la relación entre ambos.

Nuestro instrumento de evaluación se compone de varias pruebas diferentes que permiten evaluar esta competencia en diferentes contextos, de forma que se pueda reducir la posibilidad de que sea el efecto del contexto el que afecte al resultado en lugar del nivel de competencia real. Se emplean tanto problemas personales como sociales, y tecnológicos como no tecnológicos, como veremos más adelante.

En el marco PISA 2012 (OECD,2010), las pruebas discriminan varios niveles de competencia, y en este caso concreto, son seis niveles de competencia. Además, recoge la evaluación de diferentes fases de la competencia de resolución de problemas.

De esta forma, cada una de las fases presenta unos requisitos concretos para cada uno de los niveles de competencia (tabla 2) adaptados a una muestra con media de edad de 15 años, siendo la base para la creación de estas pruebas, ya que las pruebas se quieren aplicar en este target.

De las cuatro fases, las dos primeras fases se corresponden con la fase de adquisición del conocimiento, mientras que las dos segundas fases se corresponden con la fase de aplicación del conocimiento, que pueden estudiarse con alta fiabilidad desde el enfoque de sistemas complejos múltiples (Herde, Wüstenberg y Greiff, 2016).

Adquisición: La primera de las fases de la resolución de problemas corresponde con la exploración y comprensión del problema, explorar la situación para entender la relación entre los diferentes elementos o variables del problema. La segunda fase consiste en la representación y formulación del problema, de cara al establecimiento de hipótesis sobre cómo es el problema y cómo van a poder resolverlo.

Aplicación: La tercera fase de resolución de problemas es la planificación y ejecución del plan para resolver el problema, que parte de la representación que nos hayamos hecho del mismo. La cuarta y última fase es la observación y reflexión sobre si

los pasos que estamos llevando a cabo nos están conduciendo a la resolución del problema, o si por el contrario las acciones que realizamos nos alejan más de la resolución del mismo.

Para cubrir la validez de criterio en el instrumento, se han utilizado dos pruebas empleadas en la propia evaluación de PISA 2012, que han sido liberadas para todos los investigadores que deseen emplearlas en sus estudios, de forma que no todas las pruebas son de elaboración propia.

Las pruebas de PISA escogidas son *Climatizador* y *Billetes*. La primera pertenece al marco de LSE, al que pertenece MicroDYN; y la segunda al marco de FSA, al que pertenece MicroFIN.

Respecto a las pruebas creadas para la confección de nuestra batería de pruebas que conformará nuestro instrumento de evaluación, tres de estas pruebas tienen un enfoque cualitativo, relativo por tanto al enfoque de MicroFIN, y son *Vídeo*, *Agenda* e *Tren*. La cuarta prueba tiene un enfoque cuantitativo, basado en MicroDYN, y es la prueba *Coche*.

En total hay 6 preguntas del enfoque MicroFIN frente a 4 preguntas del enfoque MicroDYN, lo que reduce el posible desequilibrio que podría provocar si existiesen diferencias en los resultados de ambos enfoques.

Si, además, sumamos las preguntas de las pruebas extraídas de PISA, serían un total de 9 preguntas del enfoque MicroFIN frente a 6 preguntas del enfoque MicroDYN.

- Prueba *Climatizador* (PISA). 2 preguntas.
- Prueba *Billetes* (PISA). 3 preguntas.
- Prueba *Vídeo*. 2 preguntas.
- Prueba *Agenda*. 2 preguntas.
- Prueba *Tren*. 2 preguntas.
- Prueba *Coche*. 4 preguntas.

Tabla 2. Modelo PISA 2012 para la resolución de problemas complejos (OECD,2010), basado en el enfoque de sistemas múltiples complejos de Greiff, Wüstenberg y Funke (2012).

Nivel	Explorar y comprender	Representar y formular	Planear y ejecutar	Observar y reflexionar
1	Pueden explorar en un sentido limitado, solo cuando se han encontrado en situaciones muy parecidas antes.	Describen en parte el comportamiento de un simple dispositivo del día a día, basadas en sus observaciones de escenarios familiares.	Resuelven problemas lineales si hay una simple condición para ser resuelta y hay sólo uno o dos pasos para ser resuelta y conseguir el objetivo.	Tienden a no ser capaces de planificar con anticipación o establecer subobjetivos.
2	Pueden explorar un escenario de un problema no familiar y entender una pequeña parte.	Algunas veces, logran entender y controlar dispositivos digitales con controles no familiares.	Pueden probar una simple hipótesis que les sea dada y resolver un problema que tiene una sola restricción específica.	Pueden planificar y llevar a cabo un solo paso a la vez para lograr un subobjetivo, y tienen cierta capacidad para monitorizar todo el proceso hacia la solución.
3	Pueden explorar e inferir relaciones sencillas entre sus componentes. Con múltiples condiciones interrelacionadas, dejan una variable constante para ver el cambio en otras.	Pueden manejar información presentada en varios formatos. Pueden controlar dispositivos digitales simples, pero tienen problemas con dispositivos más complejos.	Pueden tratar con una condición, por ejemplo, para generar varias soluciones y chequear cuál de ellas satisface la condición.	Pueden idear y ejecutar pruebas para confirmar o refutar hipótesis. Ellos entienden la necesidad de planear y monitorizar el progreso, y son capaces de intentar una opción diferente si es necesario.
4	Pueden explorar el escenario de un problema bastante complejo en un sentido enfocado. Entienden los enlaces entre los componentes que requieren resolver el problema.	Pueden formular una hipótesis sobre por qué un sistema funciona mal, y describir cómo probarlo. Dominan dispositivos bastante complejos, pero no siempre lo hacen eficazmente.	Pueden probar sistemáticamente diferentes posibilidades y chequear si múltiples condiciones se satisfacen.	Pueden planear unos pocos pasos y monitorizar el progreso de sus planes. Normalmente son capaces de ajustar sus planes o reformular un objetivo en función del feedback.
5	Pueden explorar sistemáticamente el escenario de un problema complejo para conseguir entender cómo se estructura la información relevante.	Cuando se enfrenta a un dispositivo moderadamente complejo que no le es familiar, ellos responden rápidamente al feedback para controlar el dispositivo.	Para conseguir una solución, piensan de forma anticipada para encontrar la mejor estrategia que aborda todas las restricciones dadas.	Pueden ajustar inmediatamente sus planes cuando detectan dificultades inesperadas o cuando cometen errores que les sacan del camino.
6	Pueden explorar un escenario de una forma estratégica para entender toda la información del problema. La información puede ser presentada en formatos diferentes, requiriendo interpretación y la integración de partes relacionadas.	Desarrollan modelos mentales completos y de varios escenarios del problema, para resolver problemas complejos de forma eficaz. Cuando se enfrentan a tecnología compleja que no les es familiar, aprenden rápidamente cómo controlarla en un camino óptimo.	Puedes establecer hipótesis generales sobre un sistema y probarlo a fondo. Pueden guiar una premisa a través de una conclusión lógica o reconocer cuando falta información disponible para lograrlo.	Para lograr una solución, pueden crear planes complejos, flexibles y de múltiples pasos que monitorizan continuamente durante la ejecución. Cuando es necesario, ellos modifican sus estrategias, teniendo en cuenta todas las restricciones, tanto explícitas como implícitas.

Como se puede ver en la tabla 3, para crear una batería de pruebas equilibrada, cada una de las preguntas de estas pruebas evalúan una fase diferente en un nivel distinto o en varios niveles distintos según los resultados. Consideramos que no es necesario incluir pruebas del nivel 1, por lo que se han evaluado los niveles del 2 al 6 en todas las fases.

Además, en la tabla 3 se puede ver cuál es el porcentaje esperado de aciertos para cada nivel de competencia, en base a los resultados obtenidos en PISA 2012 y al marco teórico de requisitos para cada nivel en cada una de las fases (tabla 2).

Tabla 3. Relación entre las pruebas y las diferentes fases de resolución de problemas en los niveles evaluados. En naranja aparecen las preguntas de las pruebas utilizadas en PISA 2012. Fuente: elaboración propia.

ivel		Explorar y comprender	Representar y formular	Planear y ejecutar	Observar y reflexionar
	0-100	Billetes. Pregunta 2. Puntuación parcial	Agenda. Pregunta 1. Parcial	Agenda. Pregunta 2.	Vídeo. Pregunta 2. Puntuación parcial.
	0-70	Vídeo. Pregunta 1.	Climatizador. P1. Agenda P1 Compl.	Billetes. Pregunta 1	Vídeo. Pregunta 2. Puntuación completa
	0-50	Coche. Pregunta 1. Puntuación parcial.	Tren. Pregunta 1.	Climatizador P 2. P. Coche P3 Parcial	Billetes. Pregunta 3.
	0-30	Billetes. Pregunta 2. Punt. completa	Coche. Pregunta 2. Punt. parcial.	Climatizador. P 2. Punt. completa	Tren. Pregunta 2. Parcial y completa.
	-10	Coche. Pregunta 1. Punt. completa	Coche. Pregunta 2. Punt. completa	Coche. Pregunta 3.	Coche. Pregunta 4.

Para medir adecuadamente la representación y evitar la influencia de algunos factores ampliamente conocidos en la tradición de la investigación sobre la representación en resolución de problemas, se han utilizado pruebas de diferentes contextos y con temáticas bastante familiares para los alumnos, de forma que eliminamos la influencia del contenido del problema y del conocimiento previo (tabla 4).

Tabla 4. Contexto de las pruebas. Fuente: elaboración propia

Prueba	Contexto			
	Tecnológico	No tecnológico	Personal	Social
Climatizador				
Billetes				
Vídeo				
Agenda				
Tren				
Coche				

Las pruebas han sido creadas en base a este marco, por lo que consideramos que tienen una validez de constructo adecuada, la cual requiere la definición precisa del dominio que se quiere evaluar: la resolución de problemas complejos. Los niveles y las diferentes fases permiten obtener una definición precisa, ya que este marco se basa en investigaciones con larga trayectoria sobre el tema (OECD, 2010).

Los principales autores de la resolución de problemas complejos o dinámicos (Funke, 2001; ver más en Herde, Wüstenberg y Greiff, 2016), han profundizado en el campo para demostrar la suficiencia de evaluación de este modelo, que se basa en la existencia y la importancia de procesos generales de resolución de problemas, independientes de dominio (Greiff, Wüstenberg, Molnár, Fischer, Funke y Csapó, 2013).

La existencia de estos procesos generales de resolución de problemas se basa en el empleo de estrategias como la que queremos estudiar, la estrategia computacional, que es aplicable a cualquier dominio. El empleo de este tipo de estrategias independientes de dominio es especialmente útil para la resolución de problemas complejos. De esta forma, el enfoque en el que se basa el diseño del instrumento permite comprobar la efectividad del uso de la estrategia computacional en cuanto a la resolución de problemas complejos.

Por tanto, el empleo de estos procesos generales es útil en todos los contextos, y especialmente útiles para estructurar problemas en situaciones desconocidas, hacen que el diseño de esta investigación se base en un constructo ampliamente validado.

Descripción de las pruebas del instrumento

Vamos a recordar las pruebas incluidas en el instrumento, y cada pregunta.

Prueba AGENDA

Esta prueba pertenece al enfoque MicroFIN, y tiene un contexto no tecnológico y personal. El enunciado de la prueba se puede ver en la figura 29.

Esta prueba consiste en crear un calendario de estudio para los exámenes de 5 asignaturas. Por una parte, el objetivo consiste en crear todas las pegatinas necesarias para el calendario para estudiar todos los temas de cada asignatura y repasar el temario (representar y formular). Por otra parte, el objetivo será rellenar un calendario con todas las pegatinas creadas para crear el calendario de estudio (planear y ejecutar).

Figura 29. Enunciado y pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia

La interacción que se realiza con la agenda y el formato de respuesta a través de pegatinas se puede ver en la figura 2.

Pregunta 1.

Esta pregunta es de representar y formular, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 2 y 3, según la puntuación obtenida en la respuesta. El enunciado de la pregunta se puede ver en la figura 30.

Esta pregunta tiene tres puntuaciones posibles. La puntuación máxima, 2 puntos, se consigue si cogen una pegatina por cada tema de la asignatura correspondiente – la que tenga ese mismo color – Y además cogen una pegatina para repasar el temario del examen. La puntuación intermedia, 1 punto, se consigue si cogen una pegatina por cada tema de la asignatura correspondiente pero no cogen pegatina para repasar el temario de examen. La puntuación mínima, 0 puntos, se da con cualquier otra respuesta.

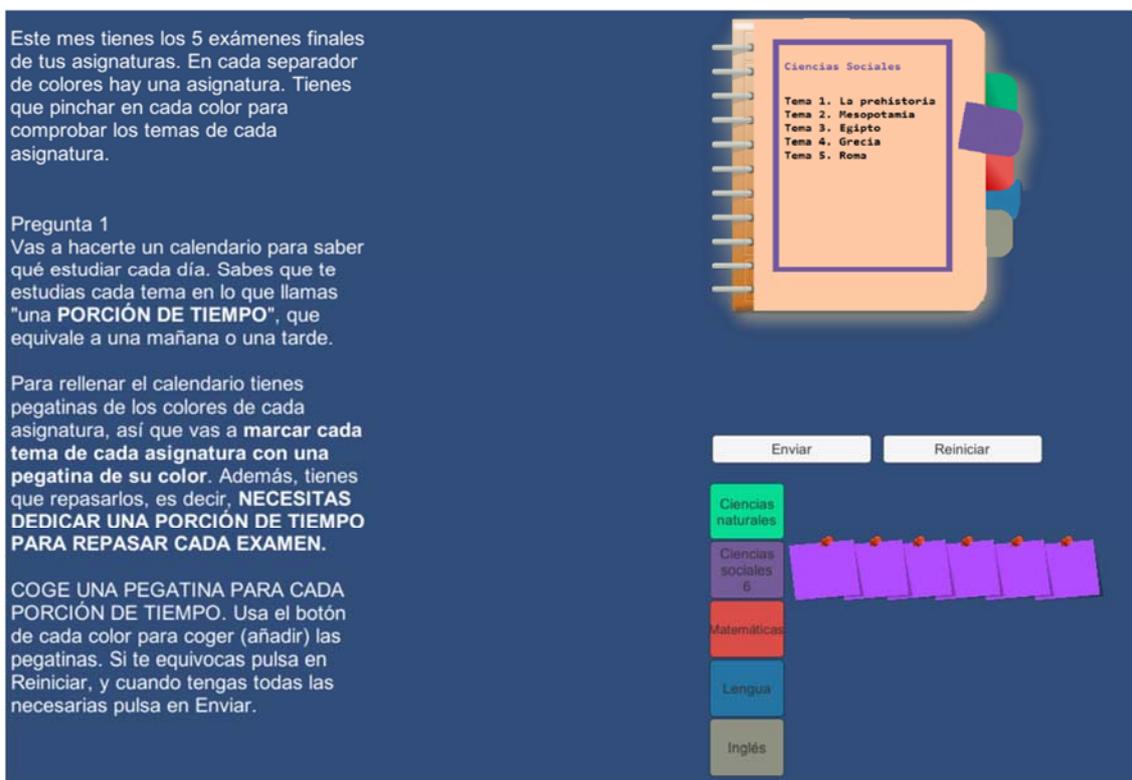


Figura 30. Interacción con la agenda y vista de las pegatinas de la pregunta 1 de la prueba Agenda, basada en el enfoque MicroFIN. Fuente: elaboración propia

Pregunta 2

Esta pregunta es de planificar y ejecutar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 2, ya que tienen que planificar todo el mes con una única restricción: la fecha de cada examen, y ejecutar un solo paso a la vez para lograr un subobjetivo (organizar el estudio cada examen). El enunciado de esta pregunta se puede ver en la figura 31.

Esta pregunta tiene dos posibles puntuaciones. El código 1 se da cuando todas las pegatinas de la asignatura están incluidas en el calendario antes del examen de esa asignatura. El código 0 se da con cualquier otra planificación.

Este mes tienes los 5 exámenes finales de tus asignaturas. En cada separador de colores hay una asignatura. Tienes que pinchar en cada color para comprobar los temas de cada asignatura.

Pregunta 2

Planifica tu mes de estudio sin contar los fines de semana. La fecha de cada examen y el turno (mañana o tarde) aparece en el calendario.

Rellena el calendario con las pegatinas, arrastrando cada pegatina a la porción de tiempo que vas a estudiar lo que pone en la pegatina. Si te equivocas, puedes arrastrar la pegatina a otro hueco. Pulsa en Enviar cuando hayas terminado

	1	2	3	4	5	6	7
M	Matemáticas Tema 1	Matemáticas Tema 1	Matemáticas Tema 1				
T	Matemáticas Tema 2	Matemáticas Tema 2	Examen Matemáticas				
M							
T		Examen Inglés					
M							
T			Examen Naturales				
M							
T			Examen Sociales				
M							
T	Examen Lengua						

Stickers available:

- Naturales Tema 1
- Naturales Tema 2
- Naturales Tema 3
- Naturales Tema 4
- Naturales Tema 5
- Naturales Tema 6
- Naturales Resumen
- Sociales Tema 1
- Sociales Tema 2
- Sociales Tema 3
- Sociales Tema 4
- Sociales Tema 5
- Sociales Resumen
- Inglés Tema 1
- Inglés Tema 2
- Inglés Tema 3
- Inglés Tema 4
- Inglés Tema 5
- Inglés Tema 6
- Inglés Resumen

Figura 31. Enunciado de la pregunta 2 de la prueba Agenda basada en el marco MicroFIN. Fuente: elaboración propia.

Prueba VÍDEO

Esta prueba pertenece al enfoque MicroFIN, y tiene un contexto no tecnológico y social. El enunciado de la prueba se puede ver en la figura 32.

En grupos de 4 tenéis que crear un vídeo para el proyecto de historia, explicando cómo era la vida en la Edad Media. La fecha de entrega es exactamente dentro de 2 semanas (14 días).

Para completar el vídeo habéis hecho una lista con todas las tareas que tenéis que hacer, y habéis creado una tabla para visualizar mejor el trabajo. Cada estrella representa una tarea (pincha sobre cada estrella para verla) y cada estrella está colocada en la fecha que debéis acabarla para que os dé tiempo a hacer las siguientes tareas.

1 ★	2	3	4	5 ★★	6	7
8	9 ★★	10 ★	11 ★★	12	13 ★★	14

Pregunta 1

Como hay poco tiempo para hacer el trabajo, tenéis que dedicar el mayor tiempo posible a cada tarea, y para ello, todos tendréis que estar siempre ocupados, es decir, **tendréis que empezar a hacer cada tarea desde que acabáis la tarea anterior.**

Entonces, ¿Qué día podremos empezar con la grabación de las escenas?

Selecciona Día

Figura 32. Enunciado y pregunta 1 de la prueba Vídeo con enfoque MicroFIN. Fuente: elaboración propia.

Esta prueba consiste en la planificación de un proyecto de vídeo para la asignatura de historia y tiene doble objetivo. Por una parte, el objetivo es que exploren las tareas que tienen que realizar para cada día para un proyecto de la clase de historia en el que tienen que grabar un vídeo, de forma que sepan el tiempo que pueden dedicar a cada tarea (explorar y comprender). Por otra parte, con esta prueba se evalúa la capacidad de identificar errores en la planificación y saber cuál es el motivo del error para poder ajustar la planificación (observar y reflexionar).

Pregunta 1.

Esta pregunta es de explorar y comprender, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 3, ya que tienen que tener en cuenta una restricción: dedicar el mayor tiempo a cada tarea, y para ello tienen que empezar al terminar las tareas anteriores. El enunciado de esta pregunta se puede ver en la figura 33.

En grupos de 4 tenéis que crear un vídeo para el proyecto de historia, explicando cómo era la vida en la Edad Media. La fecha de entrega es exactamente dentro de 2 semanas (14 días).

Para completar el vídeo habéis hecho una lista con todas las tareas que tenéis que hacer, y habéis creado una tabla para visualizar mejor el trabajo. Cada estrella representa una tarea (pincha sobre cada estrella para verla) y cada estrella está colocada en la fecha que debéis acabarla para que os dé tiempo a hacer las siguientes tareas.

1 ★	2	3	4	5 ★★	6	7
8	9 ★★	10 ★	11 ★★	12	13 ★★	14

Escena explicando la situación política.

Pregunta 1

Como hay poco tiempo para hacer el trabajo, tenéis que dedicar el mayor tiempo posible a cada tarea, y para ello, todos tendréis que estar siempre ocupados, es decir, **tendréis que empezar a hacer cada tarea desde que acabéis la tarea anterior.** Entonces, ¿Qué día podremos empezar con la grabación de las escenas?

Selecciona Día ▼

- ✓ Selecciona Día
- Día 1
- Día 2
- Día 3
- Día 4
- Día 5
- Día 6

Figura 33. Interacción con el calendario de la pregunta 1 de la prueba Vídeo, basada en el enfoque MicroFIN.
Fuente: elaboración propia

Esta pregunta cuenta con un calendario para consultar la fecha de entrega de tareas, que se muestran al pinchar sobre cada estrella. Cada estrella corresponde a una tarea. Para la respuesta, cuenta con un desplegable del 1 al 14 para que los alumnos seleccionen de dicho desplegable la solución correcta. La interacción con la prueba se puede ver en la figura 5.

Esta pregunta tiene dos puntuaciones posibles. El código 1, se da cuando responde que el día que empiezan con las grabaciones es cuando terminan la tarea anterior. Por tanto, se dará por válido que respondan tanto el día 5 como el día 6. El código 0, se da con cualquier otra respuesta.

Pregunta 2.

Esta pregunta es de observar y reflexionar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 2 y 3, según la puntuación obtenida en la respuesta. El enunciado de esta pregunta y la información que se proporciona (la propuesta de Pablo) se puede ver en la figura 34.

En el nivel 3, son capaces de monitorizar el resultado tratando con una condición, la condición de quién hace cada tarea en qué momento. En este nivel serán capaces de detectar que durante los días 6, 7 y 8; Lucía está en las dos tareas y Pablo en ninguna.

En grupos de 4 tenéis que crear un vídeo para el proyecto de historia, explicando cómo era la vida en la Edad Media. La fecha de entrega es exactamente dentro de 2 semanas (14 días). Para completar el vídeo habéis hecho una lista con todas las tareas que tenéis que hacer, y habéis creado una tabla para visualizar mejor el trabajo. Cada estrella representa una tarea (pincha sobre cada estrella para verla) y cada estrella está colocada en la fecha que debéis acabarla para que os dé tiempo a hacer las siguientes tareas.

Pregunta 2

Os ha dicho el profesor que tenéis que hacer todas las tareas en grupo, aunque sea por parejas, pero ninguna tarea puede hacerse de forma individual. Pablo ha propuesto dividir las tareas entre los 4 como aparece en la tabla de la derecha. ¿Crees que hay algún error en el reparto de tareas entre los miembros del grupo? Di brevemente dónde está el error.

Escribe aquí tu respuesta...

Enviar

1 ★	2	3	4	5 ★★	6	7
8	9 ★★	10 ★	11 ★★	12	13 ★★	14

Tarea	Día en que la tarea tiene que estar terminada	Días en los que hace la tarea	Quién hace cada tarea
Decidir qué queremos contar y cómo	1	1	Todos
Buscar información sobre la Edad Media	5	2-5	Pablo y tú
Crear una introducción al vídeo	5	2-5	Lucía y Adriana
Escena explicando la situación política	9	6-8	Lucía y tú
Escena explicando costumbres y herramientas	9	6-8	Lucía y Adriana
Escena disfrazados de gente de la época	10	9-10	Todos
Edición: cortar partes malas y tomas falsas	11	11	Pablo y tú
Ordenar escenas: secuenciar el trabajo	11	11	Lucía y Adriana
Audio: poner música de fondo	13	13	Pablo y tú
Narrar el vídeo dando sentido a las escenas	13	13	Lucía y Adriana

Figura 34. Enunciado de la pregunta 2 de la prueba Vídeo basada en el enfoque MicroFIN. Fuente: elaboración propia

En el nivel 2, tienen cierta capacidad para monitorizar el resultado, pero sólo logran entender una pequeña parte de todo el escenario del problema. En este nivel serán capaces de detectar que Lucía trabaja más que el resto o que Pablo trabaja menos que el resto, pero sin tener en cuenta la simultaneidad de las tareas de Lucía los días 6, 7 y 8.

Por tanto, esta pregunta tiene 3 posibles puntuaciones. La puntuación máxima, es decir, 2 puntos, que corresponde al nivel 3 en el que identifican dónde está el fallo de la planificación.

La puntuación intermedia, es decir, 1 punto, que corresponde al nivel 2 en el que intuyen el problema, pero no saben exactamente dónde se encuentra el fallo de la planificación. Por último, la puntuación 0 se da con cualquier otra respuesta.

Prueba BILLETES

Esta prueba pertenece al enfoque MicroDYN, y tiene un contexto tecnológico y personal. El enunciado de la prueba se puede ver en la figura 35. Para un mayor

conocimiento sobre esta prueba, ver las preguntas liberadas en Instituto Nacional de Evaluación Educativa (2013).

Pregunta 1.

Esta pregunta evalúa la fase de planear y ejecutar. Según los análisis realizados por el programa PISA, esta pregunta se situaría en el nivel 3. El enunciado de esta pregunta se puede ver en la figura 36.

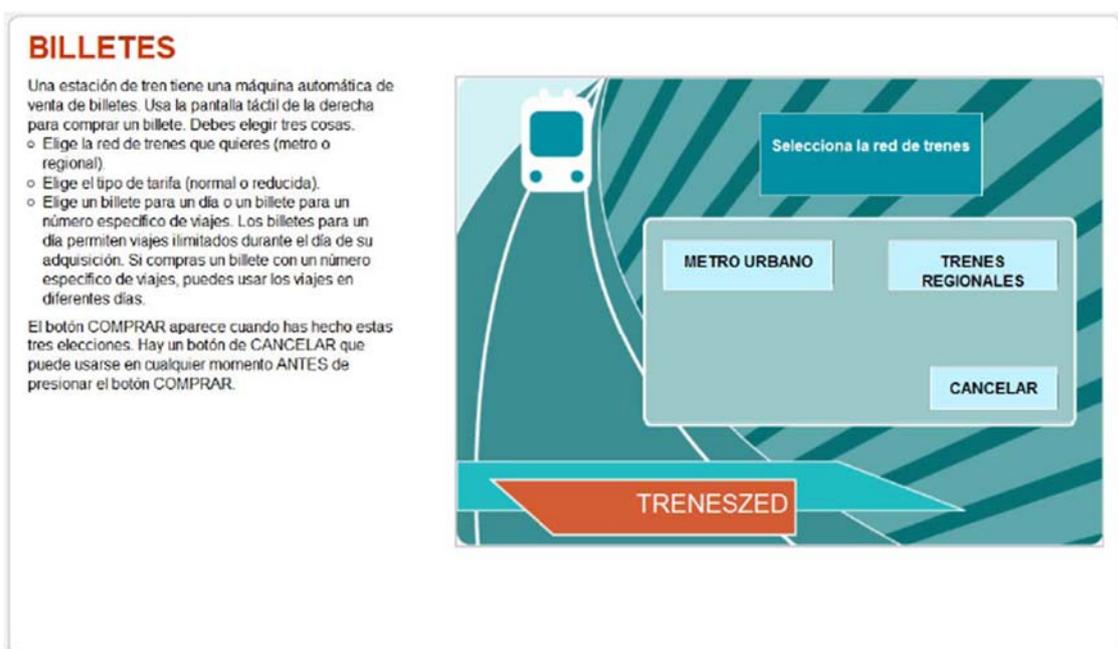


Figura 35. Enunciado de la prueba Billetes de PISA. Fuente INEE (2013)

Esta pregunta tiene dos puntuaciones posibles: 1 si han comprado el billete adecuado, y 0 si no han comprado el billete adecuado. Para que el billete sea el adecuado, es necesario que compren un billete para un tren regional, con tarifa normal, y con dos viajes individuales.



Figura 36. Enunciado de la pregunta 1 de la prueba Billetes de PISA. Fuente: INEE (2013)

Pregunta 2.

Esta pregunta evalúa la fase de explorar y comprender. Según los análisis realizados por el programa PISA, esta pregunta se situaría en los niveles 2 y 5, en función de la puntuación obtenida. El enunciado de esta pregunta se puede ver en la figura 37.

Esta pregunta tiene tres posibles puntuaciones. La puntuación máxima (2 puntos) se otorga cuando compran un billete para el metro, con una tarifa reducida y cuatro viajes individuales, y además han comparado el precio con el billete para un día en la misma tarifa, que se supone si visitan la pantalla para comprar este billete. La puntuación intermedia (1 punto) se da cuando compra cualquier billete para metro con tarifa reducida, pudiendo ser individual o diario. Pero en el caso de ser individual, no ha comparado el precio de ambos billetes, ya que si no obtendría la puntuación máxima. Para cualquier otra respuesta, la puntuación es 0.

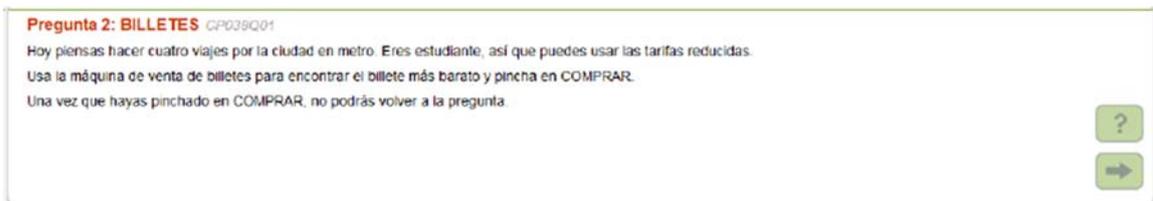


Figura 37. Enunciado de la pregunta 2 de la prueba Billetes de PISA. Fuente: INEE (2013)

Pregunta 3.

Esta pregunta evalúa la fase de observar y reflexionar. Según los análisis realizados por el programa PISA, esta pregunta se situaría en el nivel 4. El enunciado de esta pregunta se puede ver en la figura 38.

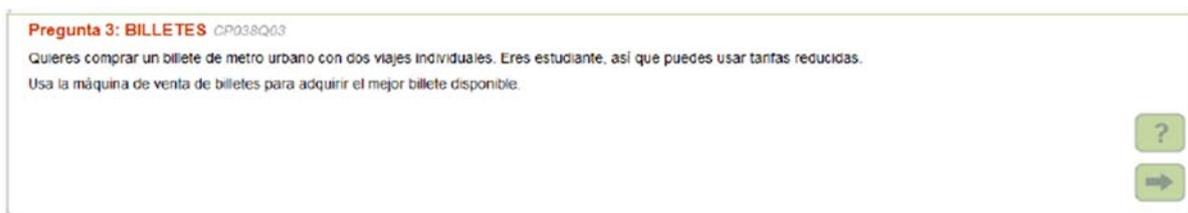


Figura 38. Enunciado de la pregunta 3 de la prueba Billetes de PISA. Fuente: INEE (2013)

En esta pregunta se encuentran con el problema de que el billete que tienen que intentar comprar no está disponible. En este caso, deberán explorar otras opciones de billetes para el metro. El billete con tarifa reducida para viajes ilimitados en el día (diario) y las opciones de billete con tarifa normal tanto para 2 viajes individuales como el billete diario. Después de comparar, deberán comprar el mejor billete.

Esta pregunta tiene dos puntuaciones: 1 punto si compras el billete disponible más adecuado, que por el precio es el billete con tarifa normal para 2 viajes individuales; y 0 puntos si compras cualquier otro billete.

Prueba TREN

Esta prueba pertenece al enfoque MicroFIN, y tiene un contexto no tecnológico y social. El enunciado de la prueba, junto a la pregunta 1, se pueden ver en la figura 38.

Esta prueba consiste en organizar un viaje en tren por Europa. Por una parte, el objetivo de la prueba es averiguar el recorrido a partir de unas anotaciones incompletas que muestran el tiempo de viaje entre ciudades (representar y formular). Por otra parte, el objetivo de la prueba consiste en identificar dos fallos cometidos en la planificación de los hoteles para el viaje (observar y reflexionar).

Pregunta 1.

Esta pregunta es de representar y formular, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 4, ya que tienen que representar el recorrido del viaje probando sistemáticamente diferentes posibilidades y chequeando si se cumplen varias condiciones simultáneas, al explorar los diferentes tiempos de viajes de cada ciudad. El enunciado de esta pregunta está en la figura 39.

(Nos vamos de viaje! Este verano vamos a hacer un interrail por Europa, viajando en tren por un montón de países. El mapa de la derecha muestra las horas que se tardan viajando entre los destinos señalados.

Pregunta 1. Ya hemos mirado los días que queremos estar en cada ciudad, y las horas de los trenes para movernos entre las ciudades. Pero por un pequeño problema al guardar el archivo, nos falta la parte en la que venían las ciudades, por lo que no sabemos el recorrido. Reconstruye el recorrido del viaje mirando el mapa a partir de las horas de viaje.

Nota: Sales desde Barcelona.

Días en la ciudad (Aproximados)	Hora salida del tren	Horas de viaje
3	9.20	0h 25min
5	12.55	1h 45min
4	9.21	2h 20min
4	7.01	0h 05min
5	8.46	4h 40min
3	8.42	4h 10min
4	19.17	15h 55 min
3	Avión a Madrid: miércoles 31. Hora 21:20	

Barcelona ▾ Elige ciudad ▾

Enviar

- Liubliana
- Londres
- Madrid
- Milán
- Niza
- París
- Praga

Figura 39. Enunciado de la prueba Tren y de la pregunta 1, prueba basada en el marco MicroFIN. Fuente: elaboración propia.

Esta pregunta tiene dos posibles puntuaciones. La puntuación 1 se da si se ha representado adecuadamente el recorrido, por lo que todas las ciudades deben estar en orden. La puntuación 0 se da con cualquier otra representación.

Pregunta 2.

Esta pregunta es de observar y reflexionar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 5, ya que en este nivel los alumnos pueden detectar errores o dificultades inesperadas para ajustar sus planes. Tanto la puntuación máxima como la

intermedia se sitúan en el nivel 5, ya que en ambos casos implica detectar un error. El enunciado de esta pregunta está en la figura 40.

¡Nos vamos de viaje! Este verano vamos a hacer un interrail por Europa, viajando en tren por un montón de países. La tabla de la derecha muestra las horas de salida más las horas de viaje, para saber la hora a la que llegamos al siguiente hotel.

Pregunta 2. Hemos hecho las reservas del hotel. Pero no estamos seguros si lo hemos reservado correctamente porque no nos hemos fijado en la hora de salida de los trenes.

Compara las fechas de los hoteles de la tabla de reservas con la tabla donde hemos apuntado los horarios de los trenes. MARCA EN LA TABLA SI HEMOS RESERVADO UNA NOCHE DE HOTEL QUE NO NECESITAMOS, Y/O SI NOS HA FALTADO POR RESERVAR ALGUNA NOCHE DE HOTEL, y cuando lo tengas Dale a Enviar. Para marcar pulsa sobre la fecha de reserva incorrecta y aparecerá un asterisco (*).

NOTA: 1 día en el hotel significa que tenemos la habitación desde las 13.00 horas de ese día hasta las 12.00 horas del día siguiente, y podemos entrar después y salir antes, respectivamente.

Enviar

Días en la ciudad (Aproximados)	Hora salida del tren	Horas de viaje
3	9.20	6h 25min
5	12.55	1h 43min
4	9.21	2h 20min
4	7.01	6h 05min
5	8.46	4h 40min
3	8.42	4h 10min
4	19:17	15h 55 min
3	Avión a Madrid: miércoles 31. Hora 21.20	

Ciudad	Día de Entrada	Día de salida
Barcelona (España)	1	4
París (Francia)	4	8 *
Bruselas (Bélgica)	9	13
Ámsterdam (Holanda)	13	17
Berlin (Alemania)	17	22
Praga (Rep. Checa)	22	25
Viena (Austria)	25	29 *
Milán (Italia)	29	31

Figura 40. Enunciado de la pregunta 2 de la prueba Tren basada en el marco MicroFIN. Fuente: elaboración propia.

Esta pregunta tiene tres posibles puntuaciones. La puntuación máxima, 2 puntos, se da cuando detectas el error de que en París te quedas el último día sin hotel (la noche del 8) y también detectas el error de que en Milán no necesitas el primer día de hotel porque pasas esa noche en el tren desde Viena (la noche del 29). La puntuación intermedia, 1 punto, se da cuando detectas un error u otro. La puntuación inferior, 0, se da si no detectas ninguno de los dos errores.

Para responder basta con marcar el día de salida de un hotel (el día 8 y el día 29) o el día de entrada al siguiente hotel (el día 9 y el día 29).

Prueba CLIMATIZADOR

Esta prueba pertenece al enfoque MicroDYN, y tiene un contexto tecnológico y personal. El enunciado de la prueba se puede ver en la figura 41. Para un mayor conocimiento sobre esta prueba, ver las preguntas liberadas en Instituto Nacional de Evaluación Educativa (2013).

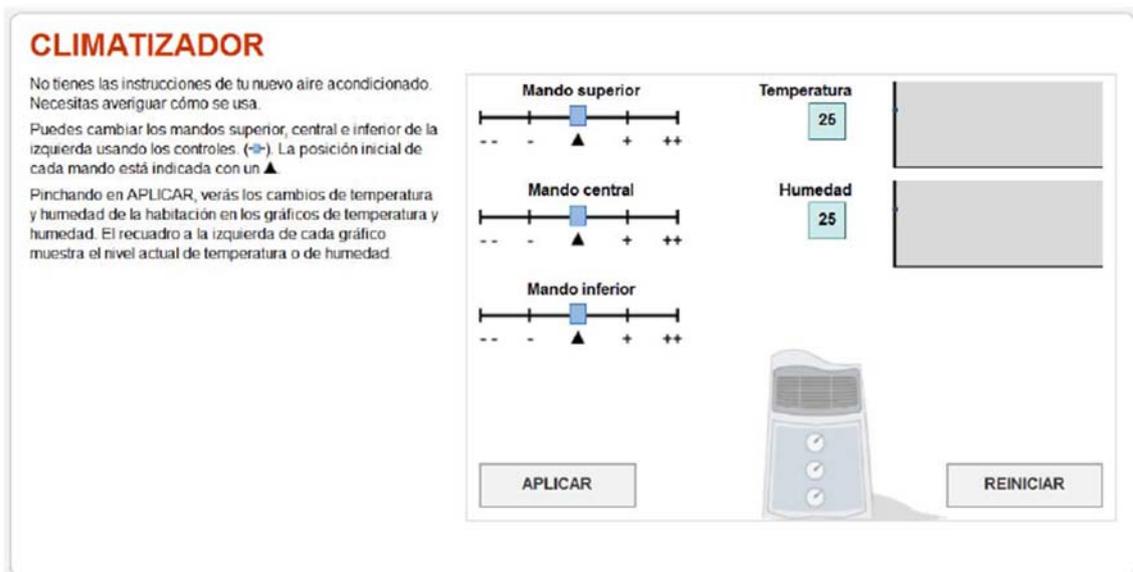


Figura 41. Enunciado de la prueba climatizador de PISA. Fuente: INEE (2013)

Pregunta 1.

La primera pregunta de Climatizador es sobre representar y formular, y según los análisis realizados por el programa PISA, esta pregunta se situaría en el nivel 3. El enunciado de esta pregunta se puede ver en la figura 42.

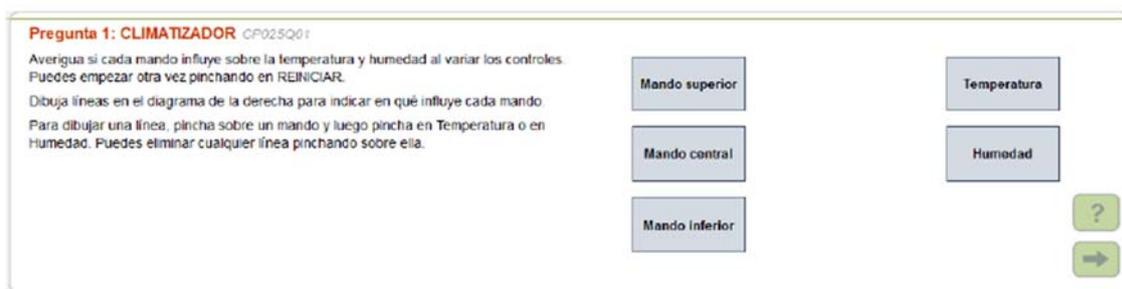


Figura 42. Enunciado de la pregunta 1 de la prueba climatizador de PISA. Fuente: INEE (2013)

Para el programa PISA, esta pregunta tenía 3 posibles puntuaciones, ya que la puntuación intermedia se otorga al haber utilizado una estrategia específica para tratar la información (vary-one-thing-at-time o VOTAT).

En nuestro caso, esta pregunta sólo tiene dos posibles puntuaciones (código 1 si es correcta, código 0 si no es correcta). Para que la pregunta se dé por correcta, la persona tiene que construir el modelo correcto, relacionando el mando superior con la temperatura, y los mandos central e inferior con la humedad. Si el modelo no es correcto, obtendrá el código 0.

Pregunta 2.

La segunda pregunta de Climatizador es sobre planear y ejecutar, y según los análisis realizados por el programa PISA, esta pregunta se situaría en los niveles 4 y 5. El enunciado de esta pregunta se puede ver en la figura 43.

Pregunta 2: CLIMATIZADOR CP025Q02

A la derecha se muestra la relación correcta entre los tres mandos, la Temperatura y la Humedad.

Usa los mandos para regular la temperatura y la humedad hasta los niveles deseados. **Hazlo en cuatro pasos como máximo.** Los niveles deseados se muestran mediante bandas rojas en los gráficos de Temperatura y Humedad. El intervalo de los valores para cada nivel deseado es 18-20 y se muestra a la izquierda de cada banda roja. **Sólo puedes pinchar en APLICAR cuatro veces y no hay botón de REINICIAR.**

Mando superior → Temperatura

Mando central → Humedad

Mando inferior → Humedad

Figura 43. Enunciado de la pregunta 2 de la prueba climatizador de PISA. Fuente: INEE (2013)

Esta pregunta tiene 3 posibles puntuaciones. La máxima puntuación, un 2, se da cuando el resultado de ambas variables está entre los valores 18 y 20. La puntuación intermedia, un 1, se da cuando el resultado no está entre los valores 18 y 20, pero la distancia a estos valores es menor de 5 en el resultado de ambas variables. Cualquier otra respuesta tiene la puntuación 0.

Prueba COCHE

Esta prueba pertenece al enfoque MicroDYN, y tiene un contexto tecnológico y personal. El enunciado de la prueba se puede ver en la figura 44.

Se trata de un sistema complejo con múltiples relaciones entre los distintos elementos del problema. Este sistema representa un coche híbrido en el que puedes elegir en todo momento el modo de conducción que deseas utilizar (híbrido, gasolina, eléctrico).

La prueba consiste en comprender la interacción entre las variables de entrada y las variables de salida del coche para llegar a una fiesta. Las variables de entrada son los modos de conducción: híbrido, gasolina y eléctrico. Las variables de salida son el progreso de avance y la energía.

El progreso de avance tiene que completarse, ya que indica que has llegado al destino, por lo que es nuestro objetivo; como cada modo de conducción avanza de modo distinto, tienes que tener en cuenta cuánto avanza cada modo para poder llegar.

Por otra parte, la energía es la que permite que puedas avanzar, por lo que digamos que es una restricción: para llegar al final tienes que tener energía, por lo que tienes que tener en cuenta cuánta energía gasta cada modo de conducción.

Además de controlar el efecto de las variables de entrada en ambas variables de salida, las variables de entrada se influyen entre ellas. Es decir, al utilizar el modo híbrido también disminuye tu modo gasolina. Al utilizar el modo gasolina, afecta también al modo eléctrico. Por último, al utilizar el modo eléctrico, influye en la variable de modo híbrido pero de una forma positiva, es decir, aumenta el modo híbrido.

Esta prueba tiene múltiples objetivos: descubrir la relación entre las variables (explorar y comprender), representar la relación (representar y formular), llegar a la fiesta – completando la barra de progreso de avance – (planear y ejecutar), e identificar cuántas veces podremos usar el modo Gasolina (observar y reflexionar).

La relación entre las variables se rige por las siguientes reglas:

1. Cada modo de conducción resta el 20% de su propia barra de progreso. Por tanto, si lo calculamos sobre 10 posiciones o unidades, utilizar un modo supone quitarle 2 unidades.
2. El progreso de avance y la energía tienen 20 posiciones o unidades. El progreso aumenta, y la energía disminuye, a diferente ritmo según el modo de conducción elegido.
3. Cada uno de los modos, influye en las demás variables en lo siguiente:
 - a. Híbrido: gasolina (-1 ud.) / energía (-3 uds.) / avanzar (+3 uds.)
 - b. Gasolina: eléctrico (-1 ud.) / energía (-2 uds.) / avanzar (+2 uds.)
 - c. Eléctrico: híbrido (+1 ud.) / energía (-1 ud.) / avanzar (+1 ud.)

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Pregunta 1
Averigua a cuáles de los elementos del problema afecta conducir el coche en modo eléctrico.

Energía
 Progreso de avance
 Gasolina
 Híbrido

Enviar

Figura 44. Enunciado de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

Pregunta 1.

Esta pregunta es de explorar y comprender, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 4 y 6, ya que la puntuación parcial equivale a comprender las

relaciones directas con las variables de resultado, temperatura y humedad, lo que correspondería a un nivel 4; y la puntuación máxima equivale a comprender además las relaciones indirectas con otras las variables que afectan a la temperatura y la humedad, lo que corresponde a un nivel 6. El enunciado de la pregunta se encuentra en la figura 45.

Esta pregunta tiene tres puntuaciones posibles. La puntuación máxima, es decir, 2 puntos, se da cuando marcan todas las relaciones: relaciones directas (energía y progreso), y relaciones indirectas (híbrido). La puntuación intermedia, es decir, 1 punto, se da cuando marcan las relaciones directas: energía y progreso. La puntuación mínima, 0 puntos, se da con cualquier otra respuesta.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Progreso de avance

Energía

Gasolina Eléctrico Híbrido

Reiniciar

Pregunta 1
Averigua a cuáles de los elementos del problema afecta conducir el coche en modo eléctrico.

Energía
 Progreso de avance
 Gasolina
 Híbrido

Enviar

Figura 45. Enunciado de la pregunta 1 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

Pregunta 2.

Esta pregunta es de representar y formular, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería a los niveles 5 y 6, ya que la puntuación intermedia implica explorar un

problema para comprender cómo se estructura la información relevante, en este caso, las relaciones directas; y la puntuación máxima implica comprender cómo se estructura toda la información del problema. El enunciado de esta pregunta aparece en la figura 46.

Esta pregunta tiene tres posibles puntuaciones. La puntuación total, 2 puntos, se consigue si representan todas las relaciones directas: energía y progreso con cada modo de conducción, y las relaciones indirectas: entre gasolina, híbrido, eléctrico; es decir, se consigue si representan adecuadamente el modelo que aparece en la figura 19. La puntuación intermedia, 1 punto, se consigue si representan sólo las relaciones directas, es decir, los 3 modos de conducción con el objetivo -progreso de avance- y con la restricción -energía-). La puntuación 0 se da con cualquier otro modelo representado.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Pregunta 2

Averigua a qué afecta cada modo de conducción. Dibuja líneas en el diagrama de abajo para indicar en qué influye cada modo de conducción.

Para dibujar una línea, pincha sobre un modo de conducción y luego pincha en Energía, Progreso de avance u otro modo de conducción. Puedes eliminar cualquier línea pinchando sobre ella.

Figura 46. Enunciado de la pregunta 2 de la prueba Coche basada en el marco MicroDYN. En negro aparecen las relaciones correctas y en rojo las otras posibles relaciones. Fuente: elaboración propia.

Pregunta 3.

Esta pregunta es de planificar y ejecutar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 6, ya que en este nivel son capaces de crear planes complejos,

flexibles y con múltiples pasos, y llevarlos a cabo monitorizando todo el proceso por si tienen que reajustar su plan. El enunciado de esta pregunta está en la figura 47.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía. NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Energía **Progreso de avance**

Pregunta 3

Intenta llegar a la fiesta. Para ello, tendrás que completar la barra de progreso de avance sin quedarte sin energía. Si no has podido terminar, pulsa enviar.

Figura 47. Enunciado de la pregunta 3 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

El nivel 6 concretamente correspondería con crear un plan eficaz, que permita llegar con el mínimo número de pasos. También pueden completar el progreso de avance sin tener el plan más eficaz, este caso correspondería al nivel 4.

Esta pregunta tiene tres posibles puntuaciones. La puntuación máxima, correspondiente a 2 puntos, se da cuando consigue llegar a la fiesta, es decir, que la barra de progreso de avance se complete, en el menor número de pasos posible, que es 8 pasos (2 pasos en modo eléctrico y 6 pasos en modo híbrido). La puntuación intermedia, correspondiente a 1 punto, se da cuando consigue llegar a la fiesta, siendo indiferente el número de pasos que haya necesitado para llegar. La puntuación mínima, correspondiente a 0 puntos, se da cuando no se llega a la fiesta porque la barra de energía se acaba antes de completar la barra de progreso.

Pregunta 4.

Esta pregunta es de observar y reflexionar, y según la definición de los niveles proporcionada en estos marcos formales de evaluación de resolución de problemas correspondería al nivel 6, ya que en este nivel son capaces de desarrollar modelos mentales con diferentes escenarios del problema y simular la ejecución de un plan para reconocer de forma anticipada si es posible lograr el objetivo o no. El enunciado de esta pregunta se muestra en la figura 48.

Esta pregunta tiene dos posibles respuestas. El código o puntuación 1, se da cuando señalan la opción 3 veces, ya que para llegar a la fiesta solo pueden usar 3 veces como máximo el modo de conducción Gasolina. El código o puntuación 0 se da con cualquier otra respuesta.

Vas a una fiesta y tienes que llevar las bebidas, así que has pensado ir en tu nuevo coche híbrido sin carnet. El problema es que se te ha olvidado cargar el coche esta noche, y tienes muy poca gasolina. Averigua qué tipo de conducción tienes que usar en cada momento (gasolina, eléctrico, híbrido) para llegar a la fiesta sin quedarte sin energía.

Para cada modo de conducción, la barra negra te indica el mínimo que debes tener para poder usarlo.

NOTA: No importa que llegues a 0 de energía porque en la fiesta hay un punto de carga.

Pregunta 4

Mi tío me ha dicho que seguramente pueda llegar en el modo Gasolina, pero yo no lo creo. Observa de nuevo cómo interacciona el sistema de modos de conducción y reflexiona sobre cuántas veces puedes utilizar **como máximo** el modo Gasolina.

0
 1
 2
 3
 4
 5

Enviar

Energía Progreso de avance

Gasolina Eléctrico Híbrido

Reiniciar

Figura 48. Enunciado de la pregunta 4 de la prueba Coche basada en el marco MicroDYN. Fuente: elaboración propia.

Procedimiento

Algunas de las pruebas elaboradas se emplearon en una prueba piloto que se realizó en formato papel durante el pasado curso 2015/2016, de cara a comprobar dudas en la interpretación de las pruebas y mejorar su redacción. Estas pruebas son las basadas en el marco de sistemas autómatas de estados finitos (FSA), es decir, las pruebas con las que se interactúa de forma cualitativa, creadas bajo el enfoque MicroFIN: agenda, vídeo y tren.

Con algunas adaptaciones en función del feedback y en base al marco en el que se fundamenta PISA 2012 para hacer de ellas pruebas interactivas, es decir, que la persona tenga que interactuar con el problema, se ha creado esta versión computarizada de las pruebas para hacer una última validación antes de utilizarlas en la investigación principal.

Algunas de estas pruebas han tenido una adaptación más sencilla que otras pruebas, debido al enfoque de la prueba o al contexto en el que se desenvuelve. Por ejemplo, las pruebas basadas en MicroFIN, es decir, las pruebas cualitativas, se pueden realizar sin necesidad de emplear un entorno computarizado, o al menos supone una mejora inferior respecto a las pruebas cuantitativas (MicroDYN) a la hora de que la persona interactúe con el problema.

De igual forma, las pruebas con un contexto no tecnológico no requieren de un entorno computarizado, o por lo menos, su mejora no es tan sustanciosa, como las pruebas con un contexto tecnológico. A partir de estas premisas, podemos saber, por ejemplo, que la prueba coche aporta mucho más en un entorno computarizado que la prueba vídeo.

En una primera fase se recogieron y analizaron los datos de la muestra que se ha utilizado exclusivamente para la investigación (28 participantes). Se realizaron unos

primeros análisis descriptivos del porcentaje de aciertos en cada pregunta para comprobar que las preguntas se sitúan en el nivel de competencia esperado (anexo 1).

Una vez se dio por suficiente una primera estadística descriptiva de las pruebas, se recogieron los datos de la segunda muestra (38 participantes) para contrastar estos análisis descriptivos entre ambas muestras (anexo 2), de cara a comprobar que no existen diferencias significativas entre ambas muestras y que la muestra total, es decir, la suma de ambas muestras, arroja resultados similares (anexo 3).

Además, de esta forma tenemos en total un mayor número de participantes ($n = 66$) que permita realizar análisis psicométricos más fiables. A continuación, se presentan los resultados obtenidos sobre la fiabilidad de las pruebas.

Resultados

Fiabilidad como consistencia interna

Para considerar que una prueba tiene una consistencia interna adecuada, debe obtenerse un Alfa de Cronbach entorno al 0,8. El análisis de la consistencia interna de las quince preguntas de estas pruebas ha obtenido un alfa de 0,656, como se puede ver en la tabla 5.

Tabla 5. Alfa de Cronbach general de las pruebas.

Estadísticas de fiabilidad	
Alfa de Cronbach	N de elementos
,656	15

Como vemos en la tabla 6, este estadístico podría aumentar, y por tanto aumentaría la consistencia interna de la batería de pruebas, si eliminamos las preguntas Agenda 1, Climatizador 2 y Coche 4.

Tabla 6. Alfa de Cronbach si se eliminase cada elemento.

Estadísticas de total de elemento					
		Media de escala si el elemento se ha suprimido	Varianz a de escala si el elemento se ha suprimido	Correla ción total de elementos corregida	Alfa de Cronbach si el elemento se ha suprimido
ge1	a	7,18	11,382	,096	,665
ge2	a	7,03	10,399	,466	,615
id1	v	7,23	10,948	,277	,639
id2	v	7,08	10,317	,321	,632
ill1	b	7,02	10,846	,322	,634
ill2	b	7,06	9,781	,349	,628
ill3	b	7,11	10,712	,348	,630
ren1	t	7,09	10,545	,404	,623
ren2	t	7,44	11,112	,215	,647
li1	c	7,06	10,735	,347	,631
li2	c	6,92	11,056	,125	,666
oche1	c	7,26	10,594	,292	,637
oche2	c	7,21	10,231	,362	,625
oche3	c	6,88	11,185	,208	,648
oche4	c	7,56	12,250	-,112	,672

Con estos resultados, se ha procedido a eliminar dichas preguntas de la prueba para comprobar el estadístico alfa de Cronbach que se alcanza al eliminarlas. Vemos en

la tabla 7 cómo se eleva el alfa de Cronbach hasta 0,672 al eliminar la pregunta Coche 4, del nivel seis de la fase de observar y reflexionar.

Tabla 7. Alfa de Cronbach al eliminar la pregunta Coche 4.

Estadísticas de fiabilidad	
Alfa de Cronbach	N de elementos
,672	14

Al eliminar la pregunta Climatizador 2, de los niveles cuatro y cinco en sus puntuaciones parcial y completa respectivamente, y correspondiente a la fase de planificar y ejecutar, se ha elevado el alfa de Cronbach a 0,685 (tabla 8).

Tabla 8. Alfa de Cronbach cuando se elimina la pregunta Climatizador 2.

Estadísticas de fiabilidad	
Alfa de Cronbach	N de elementos
,685	13

Por último, cuando eliminamos la pregunta Agenda 1, de los niveles dos y tres de la fase de representar y formular, el alfa de Cronbach alcanza el 0,699 (tabla 9).

Tabla 9. Alfa de Cronbach al eliminar la pregunta Agenda 1.

Estadísticas de fiabilidad	
Alfa de Cronbach	N de elementos
,699	12

Alcanzar un alfa de Cronbach de 0,7 es suficiente, de cara a confirmar que estas pruebas miden efectivamente resolución de problemas complejos. Después de eliminar estas preguntas, la batería de pruebas de evaluación tendrá 12 preguntas en lugar de 15 (tabla 10).

Tabla 10. Instrumento de evaluación de RCP sólo incluyendo las pruebas con las que alcanza un alfa de Cronbach de 0,7).

ivel	Explorar y comprender	Representar y formular	Planear y ejecutar	Observar y reflexionar
	Billetes. Pregunta 2.		Agenda. Pregunta 2.	Vídeo. Pregunta 2.
	Vídeo. Pregunta 1.	Climatizador. P 1.	Billetes. Pregunta 1.	Vídeo. Pregunta 2.
	Coche. Pregunta 1.	Tren. Pregunta 1.	Coche. Pregunta 3.	Billetes. Pregunta 3.
	Billetes. Pregunta 2.	Coche. Pregunta 2.		Tren. Pregunta 2.
	Coche. Pregunta 1.	Coche. Pregunta 2.	Coche. Pregunta 3.	

Discusión

En este capítulo se ha presentado una batería de pruebas, un instrumento de evaluación para la resolución de problemas complejos o dinámicos, basado en el marco que propuso el programa internacional de evaluación de estudiantes PISA en 2012 para evaluar la resolución de problemas (OECD, 2010), y que se fundamenta en el enfoque de sistemas complejos de Greiff, Wüstenberg y Funke (2012).

Esta batería de pruebas que evalúa la RPC, consta concretamente de 6 pruebas compuestas en un principio por un total por 15 preguntas. Al basarnos en un marco sólido, creemos que la validez de constructo es suficiente y no se ha considerado necesario hacer análisis sobre ésta.

Tras los análisis de fiabilidad pertinentes, de cara a aumentar la consistencia interna de las pruebas, gracias al estadístico alfa de Cronbach, se han eliminado 3 de las preguntas. De esta forma, se ha obtenido un alfa de Cronbach de 0,7. Y así, finalmente, la prueba consta de 12 preguntas.

Aunque con estas 12 preguntas no se miden todos los niveles deseados de todas las fases de resolución de problemas, son suficientes para estimar el nivel de competencia de una persona que realice estas pruebas.

Consideramos que la batería de pruebas creada tiene unas propiedades psicométricas suficientes, y que tenemos un instrumento adecuado para investigar sobre la resolución de problemas complejos en contextos educativos.

Capítulo 7. Resultados y discusión

La organización de este capítulo se basa en las tres hipótesis generales, que a su vez recogen otras hipótesis específicas, todas ellas planteadas en el capítulo en el que se define el Método de la investigación. Estas hipótesis eran:

1. El pensamiento computacional facilita la resolución de problemas complejos.
 - 1.1. El pensamiento computacional es especialmente útil para los problemas complejos de mayor nivel de dificultad.
 - 1.2. El curso de pensamiento computacional ha supuesto un desarrollo real en la capacidad de resolución de problemas complejos.
2. El pensamiento computacional facilita la adquisición de conocimiento sobre el problema, es decir, la representación del problema, que incluye la exploración y entendimiento de los elementos del problema, y la representación y formulación.
 - 2.1. El pensamiento computacional facilita especialmente la formulación.
3. El pensamiento computacional facilita el proceso de resolución del problema, es decir, el control del problema, que incluye tanto la planificación y ejecución de planes para la resolución del problema, como la observación y reflexión sobre si los planes ejecutados están acercándonos a la solución del problema.
 - 3.1. El pensamiento computacional facilita especialmente la planificación.

Antes de comprobar cada hipótesis vamos a determinar las características de la muestra de cara a elegir los estadísticos. Para ello, vamos a comprobar en primer lugar la bondad de ajuste del modelo, es decir, la normalidad de la distribución de la muestra, para utilizar un estadístico de contraste paramétrico.

Los datos se han analizado con el programa estadístico IBM SPSS Statistics, en su versión 22.

Contraste de Bondad de ajuste

Se plantea la prueba de Shapiro-Wilk de cara a contrastar la normalidad de la muestra, ya que disponemos de menos de 50 sujetos en nuestra muestra. En esta prueba, se plantea como hipótesis nula que la muestra proviene de una población que se distribuye de manera normal. Por tanto, la aceptación de la hipótesis nula quiere decir que la muestra tiene una distribución normal, mientras que el rechazo de esta hipótesis quiere decir que la muestra no sigue una distribución normal.

Para interpretar los datos que aparecen en la tabla de esta prueba, siendo la hipótesis nula que la población está distribuida normalmente, si el p-valor es menor a alfa (nivel de significancia) entonces la hipótesis nula es rechazada (se concluye que los datos no vienen de una distribución normal). Si el P-valor es mayor a alfa, no se rechaza la hipótesis y se concluye que los datos siguen una distribución normal.

Tabla 11. Prueba de Shapiro-Wilk para comprobar la bondad de ajuste.

	Pruebas de normalidad					
	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
RP_pre	,116	38	,200*	,970	38	,384
RP_pos	,113	38	,200*	,947	38	,071

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

En nuestro caso, vamos a comprobar dos distribuciones: la distribución de la muestra en el primer momento de evaluación, es decir, en el pretest; y la distribución de la muestra en el segundo momento de evaluación, es decir, en el postest.

Viendo los datos de la prueba (tabla 11), podemos confirmar que, tanto para la primera evaluación como para la segunda evaluación, la muestra tiene una distribución normal, ya que el P-valor en ambos casos (0,384 y 0,071, para ambos momentos de evaluación, respectivamente) son superiores a alfa, es decir, al nivel de significación elegido (0,05). También podemos observar la distribución de la muestra en las figuras 49 y 50, en los que se puede apreciar una distribución con tendencia a la normalidad.

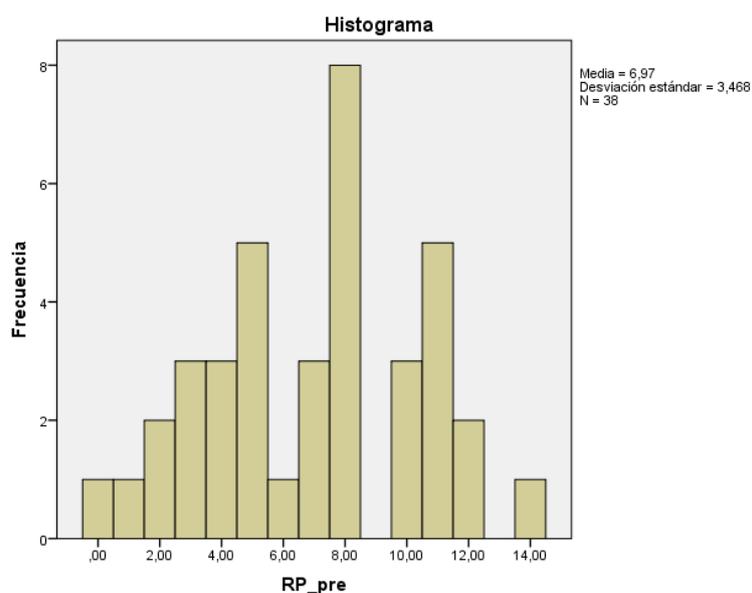


Figura 49. Histograma de la muestra en el primer momento de evaluación: pretest

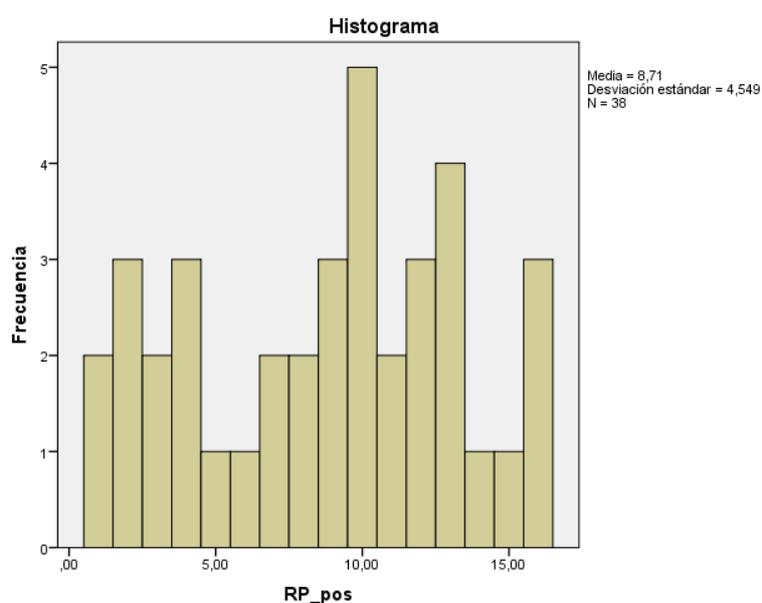


Figura 50. Histograma de la muestra en el segundo momento de evaluación: postest

Con estos resultados confirmamos que la muestra tiene una distribución normal (0,1) y por tanto debemos escoger un estadístico de contraste adecuado a esta muestra. Dado nuestro diseño de investigación, que recordemos tenía un diseño de 2x2, realizaremos la prueba T de Student para comprobar si existen diferencias entre los grupos.

Homocedasticidad de la muestra

Antes de realizar e interpretar la prueba T de Student para cada una de las hipótesis planteadas en esta investigación, vamos a comprobar en cada apartado correspondiente la homocedasticidad de las varianzas, en nuestro caso se utiliza de cara a conocer si las varianzas de las dos muestras son iguales o diferentes en cada variable planteada.

Según el resultado de la prueba, tomaremos uno u otro de los valores que nos proporciona el programa utilizado para el análisis de datos, ya que este programa hace dos cálculos en las pruebas paramétricas: un cálculo incluye el supuesto de que las varianzas son iguales, mientras que el otro cálculo incluye el supuesto de que las varianzas son diferentes.

Si el P-valor resultante de la prueba de Levene es inferior a un cierto nivel de significación (típicamente 0.05), es poco probable que las diferencias obtenidas en las variaciones de la muestra se hayan producido sobre la base de un muestreo aleatorio de una población con varianzas iguales. Por lo tanto, la hipótesis nula de igualdad de varianzas se rechaza y se concluye que hay una diferencia entre las variaciones en la población. Cuando la prueba de Levene muestra significación, se debe cambiar a pruebas generalizadas (pruebas no paramétricas), libre de supuestos de homocedasticidad.

Contraste de hipótesis estadísticas

Para realizar un correcto análisis de las hipótesis estadísticas que nos permitan conocer las diferencias entre grupos, vamos a realizar análisis diferentes. Todos ellos se van a realizar con pruebas paramétricas dados los resultados anteriores sobre la distribución de la muestra. Estas pruebas nos permitirán concluir si existen diferencias significativas entre las variables medidas.

Por una parte, los dos enfoques o aproximaciones que se realizan nos sirven para comprobar si efectivamente el grupo experimental desarrolla su capacidad para resolver problemas más que el grupo control, o algunas capacidades específicas dependiendo de la hipótesis que analicemos, comparando para ello las medias de un grupo con las medias del otro grupo. Para analizar todos los supuestos generales, utilizaremos la prueba T de Student para muestras independientes, ya que estamos comparando un grupo con otro grupo diferente; mientras que para los supuestos específicos utilizaremos el ANOVA de un factor, porque estaremos comparando más de dos medias.

Por otra parte, necesitamos comprobar si realmente podemos confirmar que ha existido un desarrollo real, de decir, un desarrollo significativo, en la capacidad de resolución de problemas, o en algunas capacidades específicas dependiendo de la hipótesis que analicemos, entre la fase previa y la fase posterior a la intervención, es decir, al curso de pensamiento computacional, en el grupo experimental. Por tanto, para confirmar que el desarrollo ha sido real, necesitamos comparar las medias entre el pretest y el posttest en el grupo experimental.

En esta línea, también comprobaremos si ha existido un desarrollo en el grupo control de forma que podamos obtener un mayor conocimiento de cara al establecimiento de conclusiones sobre el alcance del pensamiento computacional, comparando igualmente las medias del pretest con las del posttest para este grupo. Para analizar estos

supuestos de si existe un desarrollo significativo entre el momento previo (pretest) y el momento posterior (postest), utilizaremos la prueba T de Student para muestras relacionadas, ya que estamos comprobando dos medidas de un mismo grupo.

Hipótesis 1. El PC facilita la RPC

La primera hipótesis establece que el pensamiento computacional facilita la capacidad de resolución de problemas complejos, tal y como proponen desde el principio las teorizaciones sobre el PC. Para ello, comprobamos si existen diferencias significativas entre el grupo experimental (GE) y el grupo control (GC).

Para confirmar esta hipótesis, tenemos dos supuestos. El primer supuesto es que las medias de ambos grupos en la fase pretest deben ser iguales, es decir, no deben existir diferencias significativas entre grupos. El segundo supuesto es que las medias de ambos grupos en la fase postest deben ser significativamente diferentes, lo que nos llevaría a la conclusión de que gracias al curso de pensamiento computacional se ha desarrollado más la capacidad de resolución de problemas complejos.

Supuesto 1. Para la variable RP_pre: $\mu_1 = \mu_2$

Supuesto 2. Para la variable RP_pos: $\mu_1 > \mu_2$

En primer lugar, hemos obtenido las medias de ambos grupos tanto para el pretest como para el postest. Como observamos en la tabla 12, el GE (grupo 1 en las tablas de SPSS) obtuvo una media de 6,95 en el pretest frente a una media de 9,90 en el postest. Por su parte, el GC (grupo 2 en las tablas de SPSS) obtuvo una media de 7,00 en el pretest frente a una media de 7,23 en el postest.

Los datos muestran que existe mayor diferencia en el GE que en el GC, lo que nos anima a pensar que la prueba T de Student va a confirmar la primera hipótesis.

Tabla 12. Estadísticas descriptivas de los grupos en las variables RP_pre y RP_pos de la primera hipótesis de la investigación

Estadísticas de grupo					
	Grupo_cod	N	Media	Desviación estándar	Media de error estándar
RP_pre	1	21	6,9524	3,63973	,79425
	2	17	7,0000	3,35410	,81349
RP_pos	1	21	9,9048	4,79484	1,04632
	2	17	7,2353	3,86538	,93749

En cuanto al análisis inferencial, en primer lugar, debemos determinar si las varianzas se asumen iguales o no se asumen como iguales con la prueba de Levene. Como vemos en la tabla 13, ambos P-Valor (0,848 y 0,48, en el pretest y postest, respectivamente), son superiores al nivel de significación elegido (0,05). Con estos resultados asumimos varianzas iguales en ambos casos.

Tabla 13. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias

Prueba de muestras independientes										
		Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias						
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
									Inferior	Superior
RP_pre	Se asumen varianzas iguales	,037	,848	-,042	36	,967	-,04762	1,14700	-2,37384	2,27860
	No se asumen varianzas iguales			-,042	35,347	,967	-,04762	1,13693	-2,35489	2,25966
RP_pos	Se asumen varianzas iguales	,508	,480	1,857	36	,072	2,66947	1,43749	-,24589	5,58483
	No se asumen varianzas iguales			1,900	36,000	,065	2,66947	1,40488	-,17975	5,51869

Una vez sabido el cálculo estadístico que debemos elegir según los resultados de la prueba de Levene, nos fijamos en el estadístico T de Student, y en concreto en su P-valor para determinar si existen diferencias significativas entre grupos.

Como observamos en la fase pretest, el P-valor obtenido es igual a 0,967, mayor que el nivel de significación elegido de 0,05, y que al tratarse de un contraste bilateral debe ser superior a 0,025 (0,05/2), lo que confirma que de forma previa a la intervención no existían diferencias significativas entre ambos grupos.

Para la fase posttest, en la que deberían existir diferencias, el P-valor obtenido es igual a 0,072. Al ser mayor que el nivel de significación elegido, no podemos confirmar que se rechace la hipótesis nula de igualdad de medias, y que por tanto existan diferencias significativas entre los grupos.

Sin embargo, vemos cómo, mientras que el pretest existe mucha diferencia entre el P-valor y el nivel de significación; en el posttest la diferencia es mínima. Esto se podría interpretar de forma positiva, ya que, si la intervención hubiese sido más duradera, y se hubiese trabajado más el pensamiento computacional, posiblemente existiesen diferencias significativas entre ambos grupos.

Igualmente, vamos a proceder ahora a los análisis complementarios para esta primera hipótesis de la investigación.

Hipótesis 1.1. El pensamiento computacional es especialmente útil para los problemas complejos de mayor nivel de dificultad

Recordemos que el pensamiento computacional influye supuestamente en la resolución de problemas complejos. Partiendo de este supuesto, no deberían existir diferencias entre los grupos de la investigación en términos generales en cuanto a la capacidad de resolución de problemas, si no que deberían existir diferencias únicamente en lo que se refiere a problemas complejos.

Aunque todos los problemas incluidos en el modelo de evaluación creado para la investigación se pueden enmarcar como problemas complejos, el establecimiento de los niveles PISA sitúa la media de los resultados de los alumnos en el nivel 3 del modelo, de los 6 niveles totales que consta dicho modelo. Es decir, el 50% de los alumnos supera los problemas del nivel 3, y este porcentaje va disminuyendo según aumenta el nivel de los problemas.

Por tanto, vamos a realizar una distinción entre los problemas complejos situados en los niveles 2 y 3 del modelo creado, a los que denominaremos como *problemas complejos inferiores* (PCI); y los problemas complejos situados en los niveles 4, 5 y 6, a los que denominaremos *problemas complejos superiores* (PCS).

Para estudiar la relación entre PC y RPC teniendo en cuenta la dificultad propia de cada prueba, hemos dividido las preguntas de las pruebas de RPC en dos niveles, inferior y superior.

- El nivel inferior, donde se encuentran los problemas complejos inferiores (PCI) de los niveles 2 y 3 del marco utilizado, se compone de las pruebas: Billetes 2, Vídeo 1, Climatizador 1, Agenda 2, Billetes 1 y Vídeo 2. La prueba Billetes 2 está tanto en PCI (puntuación parcial) como PCS (puntuación completa). Vamos a clasificarla como PCI para obtener un equilibrio de las pruebas.
- El nivel superior incluye los problemas complejos superiores (PCS) de los niveles 4, 5 y 6 del marco empleado, y se compone de las pruebas: Coche 1, Tren 1, Coche 2, Coche 3, Billetes 3 y Tren 2.

Para confirmar esta hipótesis, partimos de los siguientes supuestos:

- (1) En los PCI no existirá diferencia entre los grupos ni antes ni después de la intervención.

Supuesto 1. Para la variable RP_inf_pre : $\mu_1 = \mu_2$

Supuesto 2. Para la variable RP_inf_pos : $\mu_1 = \mu_2$

- (2) En los PCS existirá diferencia entre los grupos después de la intervención, pero no existirá diferencia entre grupos antes de la misma puesto que las diferencias se deben precisamente a la intervención.

Supuesto 3. Para la variable RP_sup_pre : $\mu_1 = \mu_2$

Supuesto 4. Para la variable RP_sup_pos: $\mu_1 > \mu_2$

En primer lugar, se han obtenido análisis descriptivos de las variables para conocer si los datos se aproximan a nuestra hipótesis (tabla 14).

Tabla 14. Estadísticas descriptivas de los grupos en las variables RP_inf_pre, RP_inf_pos, RP_sup_pre y RP_sup_pos de la primera hipótesis de la investigación

Estadísticas de grupo					
	Grupo_cod	N	Media	Desviación estándar	Media de error estándar
RP_inf_pre	1	21	3,7143	2,10102	,45848
	2	17	3,4118	2,09341	,50773
RP_inf_pos	1	21	5,0476	2,37647	,51859
	2	17	3,7059	2,25734	,54749
RP_sup_pre	1	21	3,2381	1,97243	,43042
	2	17	3,5882	2,00184	,48552
RP_sup_pos	1	21	4,8571	2,66994	,58263
	2	17	3,5294	1,87475	,45469

Podemos ver que, el grupo experimental ha aumentado en la puntuación tanto de los PCI como de los PCS (media 3,71 en el pretest frente a un 5,04 en el postest para los problemas complejos inferiores; y media de 3,23 en el pretest frente a un 4,85 en el postest). Por su parte, el grupo control ha aumentado ligeramente en los resultados de los PCI (media 3,41 en el pretest frente a un 3,70 en el postest), pero se mantiene en los resultados de los PCS (media de 3,58 y 3,52 en pretest y postest, respectivamente).

En cuanto al contraste de hipótesis, en la tabla 15 podemos ver que se confirman la homogeneidad de varianzas para las cuatro variables.

Tabla 15. Prueba de homogeneidad de varianzas para las variables RP_inf_pre, RP_inf_pos; RP_sup_pre y RP_sup_pos del enfoque por dificultad de la primera hipótesis

Prueba de homogeneidad de varianzas				
	Estadístico de Levene	df1	df2	Sig.
RP_inf_pre	,006	1	36	,937
RP_inf_pos	,044	1	36	,835
RP_sup_pre	,142	1	36	,708
RP_sup_pos	,698	1	36	,409

Se ha realizado un ANOVA de un factor al tratarse de más de dos grupos. Los resultados del ANOVA (tabla 16) no confirman nuestra hipótesis que parecía ser apoyada con el análisis descriptivo previo de la distribución de las medias de las variables.

En los PCI no debería existir diferencia entre los grupos ni antes ni después de la intervención. Nuestro supuesto 1 comprueba las diferencias antes de la intervención, y según el P-valor de esta variable (sig. = 0,661) no existen diferencias; nuestro supuesto 2 comprueba las diferencias después de la intervención y según el P-valor (sig. = 0,085) no existen diferencias entre grupos. Por tanto, en problemas complejos inferiores no existen diferencias entre grupos, lo que estaría en línea con nuestra hipótesis.

En los PCS no debería existir diferencias entre los grupos antes de la intervención y según el P-valor (0,592) podemos confirmar que no existen diferencias entre ambos grupos. Sin embargo, debería existir diferencia entre grupos después de la intervención, pero según el P-valor (0,092) tampoco existen diferencias significativas en la capacidad de resolución de problemas complejos superiores entre ambos grupos después de la intervención.

Tabla 16. Prueba de ANOVA de un factor para comprobar si existen diferencias entre los grupos en las cuatro variables del enfoque por dificultad de la primera hipótesis

ANOVA						
		Suma de cuadrados	gl	Media cuadrática	F	Sig.
RP_inf_pre	Entre grupos	,860	1	,860	,195	,661
	Dentro de grupos	158,403	36	4,400		
	Total	159,263	37			
RP_inf_pos	Entre grupos	16,913	1	16,913	3,131	,085
	Dentro de grupos	194,482	36	5,402		
	Total	211,395	37			
RP_sup_pre	Entre grupos	1,152	1	1,152	,292	,592
	Dentro de grupos	141,927	36	3,942		
	Total	143,079	37			
RP_sup_pos	Entre grupos	16,562	1	16,562	2,999	,092
	Dentro de grupos	198,807	36	5,522		
	Total	215,368	37			

Vemos cómo los P-Valor muestran una disminución entre el pretest y el postest, pero esta disminución sucede tanto para los PCI como para los PCS.

Con estos resultados, podemos concluir que la diferenciación entre problemas complejos inferiores y problemas complejos superiores no es necesaria de cara a establecer diferencias en el éxito de resolución entre ambos grupos, ya que los resultados entre los inferiores y los superiores en el pretest son muy similares, al igual que ocurre con los resultados entre los inferiores y los superiores en el postest.

Hipótesis 1.2. El curso de pensamiento computacional ha supuesto un desarrollo real en la capacidad de resolución de problemas complejos.

Como hemos dicho, además de realizar el análisis para determinar si existen diferencias entre los grupos tanto con una aproximación general como una aproximación por dificultad, tenemos que comprobar que existe un desarrollo real significativo de la capacidad de resolución de problemas en cada uno de los grupos entre la evaluación previa y la evaluación posterior. De esta forma, podremos confirmar que los resultados obtenidos son fiables, ya que el curso de PC habrá sido eficaz.

Resultados en el grupo experimental

Primero se ha realizado una prueba de correlación entre las medidas de antes y después, para comprobar que los sujetos que puntuaron más bajo en el pretest, son los que han puntuado más bajo en el postest. Igualmente, los que puntuaron más alto, habrían puntuado más alto en el postest.

Tabla 17. Correlación de la aproximación general de RPC para el grupo experimental

Correlaciones de muestras emparejadas^a

	N	Correlación	Sig.
Par 1 RP_pre & RP_pos	21	,653	,001

a. Grupo_cod = 1

Como vemos en la tabla 17, existe una correlación de ,653 entre ambas variables (RP en el pretest y RP en el postest).

Como vemos en la tabla 18, existe una correlación un tanto inferior si dividimos la RPC entre problemas complejos inferiores y problemas complejos superiores, de 0,544 y 0,624 respectivamente.

Tabla 18. Correlación de la aproximación por dificultad de RPC para el grupo experimental

	N	Correlación	Sig.
Par 1 RP_inf_pre & RP_inf_pos	21	,544	,011
Par 2 RP_sup_pre & RP_sup_pos	21	,624	,003

a. Grupo_cod = 1

Aunque son ligeramente inferiores, siguen siendo correlaciones suficientes para determinar que se mantiene la distribución de los resultados, es decir, eliminamos la posibilidad de que estos resultados se hayan podido producir de forma aleatoria o “a boleo”.

El siguiente análisis que se ha realizado es una T de Student para muestras relacionadas nos sirve para determinar si realmente han desarrollado la capacidad para resolver problemas complejos, tanto de forma general (tabla 19) como bajo el enfoque de dificultad para PCI y PCS (tabla 20).

Tabla 19. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC gracias a la intervención (GE)

	Diferencias emparejadas					t	gl	Sig. (bilateral)
	Media	Desviación estándar	Media de error estándar	95% de intervalo de confianza de la diferencia				
				Inferior	Superior			
Par 1 RP_pre - RP_pos	-2,95238	3,66710	,80023	-4,62162	-1,28314	-3,689	20	,001

a. Grupo_cod = 1

Tabla 20. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC para PCI y PCS gracias a la intervención (GE)

		Diferencias emparejadas					t	gl	Sig. (bilateral)
		Media	Desviación estándar	Media de error estándar	95% de intervalo de confianza de la diferencia				
					Inferior	Superior			
Par 1	RP_inf_pre - RP_inf_pos	-1,33333	2,15252	,46972	-2,31315	-,35352	-2,839	20	,010
Par 2	RP_sup_pre - RP_sup_pos	-1,61905	2,10894	,46021	-2,57902	-,65907	-3,518	20	,002

a. Grupo_cod = 1

Viendo los resultados de ambas tablas, comprobamos cómo la P-valor es inferior al nivel de significatividad elegido de 0,05 en los tres casos: RPC-General (P-valor = 0,01), RPC-PCI (P-valor = 0,01) y RPC-PCS (P-valor = 0,002).

Resultados en el grupo control

En este caso se ha seguido el mismo procedimiento que para el grupo experimental, en el que primero se muestra la correlación entre los resultados del pretest y del postest de las pruebas en las variables estudiadas en esta primera hipótesis, y posteriormente se realiza una prueba T de Student para comprobar si ha habido un desarrollo de la capacidad de RPC durante el periodo de intervención.

A diferencia del grupo experimental, el grupo control no debería haber desarrollado la capacidad de RPC durante este periodo, al menos de forma significativa, ya que del modo contrario no podríamos confirmar que los resultados obtenidos se deben a que se ha desarrollado esta capacidad gracias a la intervención realizada.

En cuanto a las correlaciones, vemos en las tablas 21 y 22 cómo correlacionan de forma muy significativa las variables estudiadas para esta hipótesis, para comprobar diferencias entre el pretest y el postest. Tanto cuando se mide la RPC de forma general ($r = 0,815$), como cuando se mide bajo el enfoque de dificultad, obteniendo para PCI ($r = 0,860$) y para PCS ($r = 0,561$).

Tabla 21. Correlación de la aproximación general de RPC para el grupo control

Correlaciones de muestras emparejadas^a

	N	Correlación	Sig.
Par 1 RP_pre & RP_pos	17	,815	,000

a. Grupo_cod = 2

Tabla 22. Correlación de la aproximación por dificultad de RPC para el grupo control

Correlaciones de muestras emparejadas^a

	N	Correlación	Sig.
Par 1 RP_inf_pre & RP_inf_pos	17	,860	,000
Par 2 RP_sup_pre & RP_sup_pos	17	,561	,019

a. Grupo_cod = 2

La siguiente prueba es la T de Student para muestras relacionadas, para saber si el grupo control ha desarrollado de forma significativa la capacidad de RPC durante el periodo de la intervención (tablas 23 y 24).

Tanto para la aproximación general (P-valor = 0,672), como para la aproximación por dificultad con PCI (P-valor = 0,311) y con PCS (P-valor = 0,896); los resultados obtenidos estarían mostrando que no ha existido un desarrollo significativo de la capacidad de RPC que no haya sido debido a la intervención realizada al grupo experimental.

Tabla 23. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC que no se deba a la intervención (GC)

Prueba de muestras emparejadas^a

	Diferencias emparejadas					t	gl	Sig. (bilateral)
	Media	Desviación estándar	Media de error estándar	95% de intervalo de confianza de la diferencia				
				Inferior	Superior			
Par 1 RP_pre - RP_pos	-,23529	2,25082	,54590	-1,39256	,92197	-,431	16	,672

a. Grupo_cod = 2

Tabla 24. Prueba T de Student para muestras emparejadas para conocer si hay un desarrollo significativo de la capacidad de RPC para PCI y PCS que no se deba a la intervención (GC)

		Prueba de muestras emparejadas ^a				t	gl	Sig. (bilateral)	
		Diferencias emparejadas			95% de intervalo de confianza de la diferencia				
		Media	Desviación estándar	Media de error estándar	Inferior				Superior
Par 1	RP_inf_pre - RP_inf_pos	-,29412	1,15999	,28134	-,89053	,30230	-1,045	16	,311
Par 2	RP_sup_pre - RP_sup_pos	,05882	1,81902	,44118	-,87643	,99408	,133	16	,896

a. Grupo_cod = 2

Con todos los análisis realizados para la primera de las hipótesis, podemos intuir que existe relación entre el pensamiento computacional y la resolución de problemas complejos, pero no podemos confirmar que esta relación sea significativa en términos generales.

Aunque los resultados no nos permitan confirmar esta primera hipótesis, éstos sugieren que posiblemente esta relación se dé de forma más clara con una intervención mayor a la que hemos realizado en esta investigación. Un curso de más horas sobre pensamiento computacional podría conllevar a obtener diferencias significativas entre ambos grupos, lo cual se puede considerar como una clara limitación de este estudio.

Para contrarrestar este problema, las pruebas complementarias realizadas en las hipótesis específicas 1.1 y 1.2, nos ayudan a tener mayor conocimiento sobre esta primera hipótesis. Por una parte, la hipótesis 1.1 profundiza más en el conocimiento sobre la relación de ambas variables, al contrastar niveles de dificultad dentro de la complejidad que ya suponen los problemas complejos o dinámicos. En este caso no se ha confirmado tal supuesto.

Por otra parte, la hipótesis 1.2 confirma que el instrumento de intervención utilizado: el curso de pensamiento computacional, ha impactado realmente en los alumnos. Esto nos anima a continuar analizando en qué fases el PC tiene mayor impacto.

Hipótesis 2. El PC facilita la adquisición del conocimiento sobre el problema

Nuestra segunda hipótesis quiere comprobar si existe relación entre el pensamiento computacional y la adquisición de conocimiento sobre el problema. La adquisición de conocimiento se corresponde con la representación del problema que se ha venido estudiando en el enfoque de sistemas múltiples complejos para la resolución de problemas complejos. Esto quiere decir que, de las fases de RPC que se han establecido para la evaluación de PISA 2012, incluye tanto la exploración y comprensión del problema como la representación y formulación del mismo.

Para confirmar esta hipótesis, tenemos dos supuestos. El primer supuesto es que las medias de ambos grupos en la fase pretest deben ser iguales, es decir, no deben existir diferencias significativas entre grupos. El segundo supuesto es que las medias de ambos grupos en la fase postest deben ser significativamente diferentes, lo que nos llevaría a la conclusión de que gracias al curso de PC se ha desarrollado más la adquisición en la RPC.

Supuesto 1. Para la variable Adq_pre: $\mu_1 = \mu_2$

Supuesto 2. Para la variable Adq_pos: $\mu_1 > \mu_2$

Esta variable de Adquisición, recoge las pruebas de exploración y comprensión: Billetes 2, Vídeo 1, Coche 1, y las pruebas de representación y formulación: Climatizador 1, Tren 1 y Coche 2. La variable Adq_pre recoge los resultados del pretest y la variable Adq_pos recoge los resultados de estas preguntas en el postest.

En primer lugar, hemos obtenido las medias de ambos grupos tanto para el pretest como para el postest (tabla 25). Como observamos en la tabla 15, el GE (grupo 1) obtuvo una media de 3,23 en el pretest frente a una media de 4,9 en el postest. Por su parte, el GC (grupo 2) obtuvo una media de 3,23 en el pretest y una media de 3,52 en el postest.

Tabla 25. Estadísticas descriptivas de los grupos en las variables Adq_pre y Adq_pos de la segunda hipótesis de la investigación

Estadísticas de grupo					
	Grupo_cod	N	Media	Desviación estándar	Media de error estándar
Adq_PRE	1	21	3,2381	2,18872	,47762
	2	17	3,2353	2,19458	,53226
Adq_POS	1	21	4,9048	2,73687	,59723
	2	17	3,5294	2,57676	,62496

Vemos cómo las puntuaciones en el GE mejoran en mayor medida que en el GC, y lo que es más significativo, vemos cómo ambos grupos parten de la misma puntuación.

Para el análisis inferencial de esta hipótesis, en primer lugar, debemos determinar si las varianzas se asumen iguales o no se asumen como iguales con la prueba de Levene. Como vemos en la tabla 26, ambos P-Valor (0,828 y 0,895, en el pretest y posttest, respectivamente), son superiores al nivel de significación elegido (0,05). Con estos resultados asumimos varianzas iguales en ambos casos.

Tabla 26. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias para la segunda hipótesis

Prueba de muestras independientes										
		Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias					95% de intervalo de confianza de la diferencia	
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	Inferior	Superior
Adq_PRE	Se asumen varianzas iguales	,048	,828	,004	36	,997	,00280	,71493	-1,44715	1,45275
	No se asumen varianzas iguales			,004	34,333	,997	,00280	,71514	-1,45002	1,45562
Adq_POS	Se asumen varianzas iguales	,018	,895	1,581	36	,123	1,37535	,87009	-,38928	3,13998
	No se asumen varianzas iguales			1,591	35,129	,121	1,37535	,86444	-,37933	3,13003

Una vez sabido el cálculo estadístico que debemos elegir según los resultados de la prueba de Levene, nos fijamos en el estadístico T de Student, y más concretamente en su P-valor para determinar si podemos considerar que existen diferencias significativas entre los grupos (tabla 26).

Como observamos en la fase pretest, el P-valor obtenido es igual a 0,997, mayor que el nivel de significación elegido de 0,05, lo que confirma que de forma previa a la intervención no existían diferencias significativas entre ambos grupos.

Para la fase posttest, en la que deberían existir diferencias, el P-valor obtenido es igual a 0,123. Al ser mayor que el nivel de significación elegido, no podemos confirmar que se rechace la hipótesis nula de igualdad de medias, y que por tanto existan diferencias significativas entre los grupos.

Sin embargo, vemos cómo en el pretest existe una diferencia mucho mayor que en el posttest entre el P-valor y el nivel de significación. Esto se podría interpretar de forma positiva, ya que, si la intervención hubiese sido más duradera, posiblemente existiesen diferencias significativas entre ambos grupos.

Hipótesis 2.1. El pensamiento computacional facilita especialmente la formulación.

Nuestra segunda hipótesis también pretende indagar sobre la relación concreta en el pensamiento computacional y la fase de representación y formulación en la resolución de problemas complejos, es decir, si facilita la representación y formulación de éstos de cara a facilitar el proceso de resolución.

Para confirmar esta hipótesis, tenemos dos supuestos. El primer supuesto es que las medias de ambos grupos en la fase pretest deben ser iguales, es decir, no deben existir diferencias entre grupos. El segundo supuesto es que las medias de ambos grupos en la fase posttest deben ser significativamente diferentes, lo que nos llevaría a la conclusión de que gracias al curso de pensamiento computacional se ha desarrollado más la capacidad de representación en la resolución de problemas complejos.

Supuesto 1. Para la variable Repres_pre: $\mu_1 = \mu_2$

Supuesto 2. Para la variable Repres_pos: $\mu_1 > \mu_2$

Esta variable de Representación, recoge únicamente las pruebas Climatizador 1, Tren 1 y Coche 2, a diferencia de la hipótesis de adquisición de conocimiento. Para la variable Repres_pre se recogen los resultados del pretest y para la variable Repres_pos se recogen los resultados de estas preguntas en el postest.

En primer lugar, hemos obtenido las medias de ambos grupos tanto para el pretest como para el postest (tabla 27). Como observamos en la tabla 27, el GE (grupo 1) obtuvo una media de 1,76 en el pretest frente a una media de 2,47 en el postest. Por su parte, el GC (grupo 2) obtuvo una media de 1,76 en el pretest y una media de 2,00 en el postest.

Tabla 27. Estadísticas descriptivas de los grupos en las variables Repres_pre y Repres_pos de la segunda hipótesis de la investigación

Estadísticas de grupo					
	Grupo_cod	N	Media	Desviación estándar	Media de error estándar
Repres_PRE	1	21	1,7619	1,22085	,26641
	2	17	1,7647	1,20049	,29116
Repres_POS	1	21	2,4762	1,36452	,29776
	2	17	2,0000	1,45774	,35355

Se aprecia claramente cómo las puntuaciones en el GE mejoran en mayor medida que en el GC, ya que ambos grupos parten de la misma puntuación en el pretest. A continuación, procedemos al análisis inferencial de la hipótesis 2.1.

En primer lugar, debemos determinar si las varianzas se asumen iguales o no se asumen como iguales con la prueba de Levene. Como vemos en la tabla 28, ambos P-Valor (0,905 y 0,837, en el pretest y postest, respectivamente), son superiores al nivel de significación elegido (0,05). Con estos resultados asumimos varianzas iguales en ambos casos.

Tabla 28. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias para la hipótesis 2.1

		Prueba de muestras independientes								
		Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias					95% de intervalo de confianza de la diferencia	
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	Inferior	Superior
Repres_PRE	Se asumen varianzas iguales	,014	,905	-,007	36	,994	-,00280	,39537	-,80465	,79905
	No se asumen varianzas iguales			-,007	34,603	,994	-,00280	,39465	-,80432	,79871
Repres_POS	Se asumen varianzas iguales	,043	,837	1,038	36	,306	,47619	,45895	-,45460	1,40698
	No se asumen varianzas iguales			1,030	33,332	,310	,47619	,46224	-,46388	1,41626

Una vez sabido el cálculo estadístico que debemos elegir según los resultados de la prueba de Levene, nos fijamos en el estadístico T de Student, y más concretamente en su P-valor para determinar si podemos considerar que existen diferencias significativas entre los grupos.

Como observamos en la fase pretest, el P-valor obtenido es igual a 0,994, mayor que el nivel de significación elegido de 0,05, lo que confirma que de forma previa a la intervención no existían diferencias significativas entre ambos grupos.

Para la fase posttest, en la que deberían existir diferencias, el P-valor obtenido es igual a 0,306. Al ser mayor que el nivel de significación elegido, no podemos confirmar que se rechace la hipótesis nula de igualdad de medias, y que por tanto existan diferencias significativas entre los grupos.

Al igual que en la adquisición de conocimiento, en el pretest existe una diferencia mucho mayor que en el posttest entre el P-valor y el nivel de significación.

Viendo los resultados de esta segunda hipótesis, nos encontramos en una situación similar a la sucedida en la primera hipótesis. Aunque los datos no nos permitan confirmar que existen diferencias ni en general (adquisición) ni en particular (representación/formulación), los resultados sugieren que esta relación existe y que con una mayor intervención encontraríamos diferencias entre ambos grupos.

Hipótesis 3. El PC facilita el proceso de resolución del problema

Nuestra tercera hipótesis quiere comprobar si existe relación entre el pensamiento computacional y la aplicación de conocimiento sobre el problema, es decir, el proceso de resolución del problema. La aplicación de conocimiento corresponde al control del sistema que se ha venido estudiando en el enfoque de sistemas múltiples complejos para la resolución de problemas complejos. Esto quiere decir que, de las fases de RPC que se han establecido para la evaluación de PISA 2012, incluye tanto la planificación y ejecución como la observación y reflexión sobre el éxito de la resolución.

Para confirmar esta hipótesis, tenemos dos supuestos. El primer supuesto es que las medias de ambos grupos en la fase pretest deben ser iguales, es decir, no deben existir diferencias significativas entre grupos. El segundo supuesto es que las medias de ambos grupos en la fase postest deben ser significativamente diferentes, lo que nos llevaría a la conclusión de que gracias al curso de pensamiento computacional se ha desarrollado el control del problema como parte de la resolución de problemas complejos.

Supuesto 1. Para la variable Adq_pre : $\mu_1 = \mu_2$

Supuesto 2. Para la variable Adq_pos : $\mu_1 > \mu_2$

Esta variable de Aplicación, recoge las pruebas de planificación y ejecución: Agenda 2, Billetes 1 y Coche 3, y las pruebas de observación y reflexión: Vídeo 2, Billetes 3 y Tren 2. La variable $Aplic_pre$ recoge los resultados del pretest y la variable $Aplic_pos$ recoge los resultados de estas preguntas en el postest.

En primer lugar, hemos obtenido las medias de ambos grupos tanto para el pretest como para el postest (tabla 29). Como observamos en la tabla 19, el GE (grupo 1) obtuvo una media de 3,71 en el pretest frente a una media de 5,00 en el postest. Por su parte, el GC (grupo 2) obtuvo una media de 3,76 en el pretest y una media de 3,70 en el postest.

Tabla 29. Estadísticas descriptivas de los grupos en las variables *Aplic_pre* y *Aplic_pos* de la tercera hipótesis de la investigación

Grupo_cod	N	Media	Desviación estándar	Media de error estándar
Aplic_PRE 1	21	3,7143	1,76473	,38510
2	17	3,7647	1,71499	,41595
Aplic_POS 1	21	5,0000	2,56905	,56061
2	17	3,7059	1,68689	,40913

Los datos muestran una mejora en las puntuaciones en el GE mientras que en el GC las puntuaciones empeoran muy ligeramente.

Para el análisis inferencial de esta hipótesis, en primer lugar, debemos determinar si las varianzas se asumen iguales o no se asumen como iguales con la prueba de Levene. Como vemos en la tabla 20, el P-Valor del pretest es 0,877, superior al nivel de significación elegido (0,05), mientras que el P-Valor del posttest es 0,026, inferior al nivel de significación elegido (0,05). Con estos resultados asumimos varianzas iguales en el pretest, pero no podemos asumir que las varianzas sean iguales en el posttest.

Tabla 30. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias para la tercera hipótesis

	Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias						
	F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
								Inferior	Superior
Aplic_PRE	,024	,877	-,089	36	,930	-,05042	,56860	-1,20359	1,10275
Se asumen varianzas iguales			-,089	34,756	,930	-,05042	,56684	-1,20146	1,10062
Aplic_POS	5,422	,026	1,786	36	,082	1,29412	,72451	-,17525	2,76348
Se asumen varianzas iguales			1,865	34,680	,071	1,29412	,69403	-,11530	2,70353

Una vez sabido el cálculo estadístico que debemos elegir según los resultados de la prueba de Levene, nos fijamos en el estadístico T de Student, y más concretamente en su P-valor correspondiente para cada caso, para determinar si podemos considerar que existen diferencias significativas entre los grupos (tabla 30).

Como observamos en la fase pretest, el P-valor obtenido es igual a 0,930, mayor que el nivel de significación elegido de 0,05, lo que confirma que de forma previa a la intervención no existían diferencias significativas entre ambos grupos.

Para la fase postest, en la que deberían existir diferencias, el P-valor obtenido es igual a 0,071. Al ser mayor que el nivel de significación elegido, no podemos confirmar que se rechace la hipótesis nula de igualdad de medias, y que por tanto existan diferencias significativas entre los grupos.

Sin embargo, vemos que el P-Valor del postest está muy cerca del nivel de significación, con una diferencia de 0,02. Esto se podría interpretar de forma muy positiva, ya que, si la intervención hubiese sido más duradera, seguramente existieran diferencias significativas entre ambos grupos.

Hipótesis 3.1. El pensamiento computacional facilita especialmente la planificación.

Nuestra tercera hipótesis también pretende indagar sobre la relación concreta en el pensamiento computacional y la fase de planificación y ejecución de los planes en la resolución de problemas complejos, es decir, si facilita concretamente la elección de estrategias y mejora el proceso de resolución de los problemas complejos.

Para confirmar esta hipótesis, tenemos dos supuestos. El primer supuesto es que las medias de ambos grupos en la fase pretest deben ser iguales, es decir, no deben existir diferencias significativas entre grupos. El segundo supuesto es que las medias de ambos grupos en la fase postest deben ser significativamente diferentes, lo que nos llevaría a la conclusión de que gracias al curso de pensamiento computacional se ha desarrollado más la capacidad de planificación en la resolución de problemas complejos.

Supuesto 1. Para la variable Plan_pre: $\mu_1 = \mu_2$

Supuesto 2. Para la variable Plan_pos: $\mu_1 > \mu_2$

Esta variable de Planificación, recoge las pruebas Agenda 2, Billetes 1 y Coche 3, para la variable Plan_pre se recogen los resultados del pretest y para la variable Plan_pos se recogen los resultados de estas preguntas en el postest.

En primer lugar, hemos obtenido las medias de ambos grupos tanto para el pretest como para el postest (tabla 31). Como observamos en la tabla 15, el GE (grupo 1) obtuvo una media de 2,38 en el pretest frente a una media de 2,47 en el postest. Por su parte, el GC (grupo 2) obtuvo una media de 2,17 en el pretest frente a una media de 2,05 en el postest. Los datos muestran una mejora en las puntuaciones en el GE mientras que en el GC las puntuaciones empeoran.

Tabla 31. Estadísticas descriptivas de los grupos en las variables Plan_pre y Plan_pos de la segunda hipótesis de la investigación

Estadísticas de grupo					
	Grupo_cod	N	Media	Desviación estándar	Media de error estándar
Plan_PRE	1	21	2,3810	,97346	,21243
	2	17	2,1765	,88284	,21412
Plan_POS	1	21	2,4762	,92839	,20259
	2	17	2,0588	,89935	,21812

Ahora procedemos al análisis inferencial de la segunda hipótesis:

En primer lugar, debemos determinar si las varianzas se asumen iguales o no se asumen como iguales con la prueba de Levene. Como vemos en la tabla 32, ambos P-Valor (0,632 y 0,643, en el pretest y postest, respectivamente), son superiores al nivel de significación elegido (0,05). Con estos resultados asumimos varianzas iguales en ambos casos.

Tabla 32. Prueba de Levene para la homocedasticidad de las varianzas y prueba T de Student para la igualdad de medias

		Prueba de muestras independientes									
		Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias						95% de intervalo de confianza de la diferencia	
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	Inferior	Superior	
Plan_PRE	Se asumen varianzas iguales	,234	,632	,671	36	,507	,20448	,30481	-,41370	,82267	
	No se asumen varianzas iguales			,678	35,491	,502	,20448	,30162	-,40753	,81649	
Plan_POS	Se asumen varianzas iguales	,218	,643	1,397	36	,171	,41737	,29872	-,18846	1,02319	
	No se asumen varianzas iguales			1,402	34,796	,170	,41737	,29769	-,18711	1,02184	

Una vez sabido el cálculo estadístico que debemos elegir según los resultados de la prueba de Levene, nos fijamos en el estadístico T de Student, y más concretamente en su P-valor para determinar si existen diferencias entre los grupos.

Como observamos en la fase pretest, el P-valor obtenido es igual a 0,507, mayor que el nivel de significación elegido de 0,05, lo que confirma que de forma previa a la intervención no existían diferencias significativas entre ambos grupos.

Para la fase posttest, en la que deberían existir diferencias, el P-valor obtenido es igual a 0,171. Al ser mayor que el nivel de significación elegido, no podemos confirmar que se rechace la hipótesis nula de igualdad de medias, y que por tanto existan diferencias significativas entre los grupos.

Sin embargo, vemos cómo, mientras que el pretest existe una diferencia considerable entre el P-valor y el nivel de significación; en el posttest la diferencia se reduce. Esto se podría interpretar de forma positiva, ya que, si la intervención hubiese sido más duradera, y se hubiese trabajado más el pensamiento computacional, posiblemente existiesen diferencias significativas entre ambos grupos.

Por tanto, podemos concluir que el pensamiento computacional está relacionado con el proceso de resolución de problemas, y que debe demostrarse con una intervención mayor a la llevada a cabo.

Discusión de los resultados

Para finalizar este capítulo, la tabla 33 recoge los resultados de todas las pruebas hechas para el contraste de medias, para conocer en qué fases influye más el PC.

Al no haber obtenido diferencias significativas en ninguna de nuestras hipótesis, vamos a comparar el P-Valor de cada uno de nuestros análisis, determinando que cuanto mayor sea la diferencia entre este P-Valor en el pretest y el posttest, mayor habrá sido la influencia del pensamiento computacional, ya que habría mayor diferencia entre los grupos. Además, vamos a observar la diferencia entre el P-Valor final, es decir, el P-Valor del posttest y el nivel de significación (0,05), concluyendo que cuanto más cerca se encuentren ambos valores, mayor diferencia habrá entre los grupos.

Observamos que la mayor diferencia se da en la RPC en general (0,895), seguida de la adquisición (0,874) y de la aplicación (0,859). Con los resultados del posttest, igualmente vemos cómo el pensamiento computacional influye en la RPC general (0,072), y en la aplicación (0,071). En menor medida, vemos cómo también influye en la adquisición del conocimiento (0,123).

Tabla 33. Resumen de los P-Valor obtenidos para las diferentes hipótesis en relación con las fases de RPC, tanto en el pretest como en el posttest. Fuente: elaboración propia

FASE DE RPC	P-VALOR PRETEST	P-VALOR POSTEST
RPC	0,967	0,072
Adquisición del conocimiento del problema	0,997	0,123
Representación y formulación	0,994	0,306
Aplicación del conocimiento del problema	0,930	0,071
Planificación y ejecución	0,507	0,171

Con estos resultados podemos confirmar la relación entre ambas variables, especialmente su relación con el proceso de resolución, a pesar de que intervención no haya sido suficiente para mostrar diferencias significativas.

La escasa diferencia entre grupos se ha dado en los procesos comunes del pensamiento computacional, que recordemos incluyen el tratamiento de datos, la abstracción y la creación del algoritmo, y que han sido medidos a través de las pruebas de resolución de problemas complejos creadas.

Esto podría deberse a que estos procesos son más complejos de desarrollar que los procesos propios del pensamiento computacional, que recordemos eran descomposición o modularización, automatización, paralelismo y simulación.

El pensamiento computacional tiene hondas raíces en habilidades generales, difíciles de desarrollar. Esto serviría como explicación a por qué los estudios que se han propuesto para evaluar el pensamiento computacional, revisados en el capítulo 4, obtienen mejores resultados, ya que estos estudios evalúan habilidades muy concretas ligadas a aspectos instruccionales específicos.

Por el contrario, los estudios que evalúan habilidades generales obtienen escasos resultados, probablemente porque los procesos de aprendizaje sean muchos más lentos al tratarse de habilidades generales.

Bloque 3. Conclusiones generales

Capítulo 8. Conclusiones y futuras líneas de investigación

Esta investigación ha tenido como objetivo principal conocer la estructura del pensamiento computacional de cara a comprender cómo se relaciona este pensamiento con la resolución de problemas complejos, concretamente qué fases de la resolución de problemas complejos se ven mejoradas con el empleo de un pensamiento computacional. En nuestro estudio hemos comprobado cómo existe una relación entre el pensamiento computacional y la resolución de problemas, tanto en la representación del problema como en el proceso de resolución del mismo.

Estos resultados son importantes por la necesidad de conocer las ventajas de adoptar este nuevo estilo de pensamiento en la resolución de problemas, de modo que realmente aporte valor y sepamos aprovechar sus ventajas, para lo que tenemos que conocer cuáles son realmente sus ventajas en la resolución de problemas, es decir, cuándo aporta valor, al igual que sostenían Blackwell, Church y Green (2008).

A partir de nuestro objetivo hemos podido proponer una definición en términos operativos de este pensamiento, a partir de todo lo que se ha escrito hasta ahora sobre el pensamiento computacional. Esta definición nos facilita su evaluación de forma fiable, y nos acerca a las claves sobre cómo mejorarlo y enseñarlo en las escuelas como demandan, entre otros, Hu (2011) y Selby y Woollard (2013), claves que se comentan más adelante.

Para crear esa definición operativa, se ha partido de la revisión que realizaron Selby y Woollard (2013) de todas las propuestas de la literatura, para comprobar los procesos de pensamiento que más se repiten, y distinguirlos de aspectos de programación. Esta revisión coincide casi en su totalidad con la propuesta que ya había realizado la CSTA y la ISTE (2011), en su intento por definir este pensamiento de forma operativa para mejorar su enseñanza y desarrollarlo en los alumnos.

Como recordamos, estos procesos son recopilación de datos, análisis de datos, representación de datos, abstracción, creación de un algoritmo, descomposición del problema, automatización, paralelismo y simulación, y son los que han tomado como base para esta investigación.

El hecho de que algunos de los procesos definidos en el pensamiento computacional sean comunes a la resolución de problemas y otros sean procesos habituales y característicos de las Ciencias de la Computación, nos ha llevado a establecer un marco de estudio, para los procesos del pensamiento computacional en el que se distinguen procesos propios del pensamiento computacional, los que usarían habitualmente los científicos de la computación a diferencia de otras disciplinas, y procesos comunes a la RPC, que nos indican que este pensamiento permite mejorar estos procesos de resolución de problemas, que en este caso son comunes a todas las disciplinas. Este marco o modelo para el análisis, se puede observar en la figura 51.

Por una parte, encontramos los procesos compartidos entre PC y RP, y que son los relacionados con el tratamiento de datos: recopilación, análisis y representación, además del proceso de abstracción y la creación del algoritmo. Estos procesos siempre aparecen en la resolución de problemas, empleemos una estrategia computacional o cualquier otra estrategia. Estos procesos se han medido con las pruebas de resolución de problemas creadas. Los procesos de abstracción y recopilación, análisis y representación de los datos se han medido en la adquisición del conocimiento o representación del problema; mientras que el proceso de creación del algoritmo se ha medido en la aplicación del conocimiento o proceso de resolución del problema.

Por otra parte, están los procesos propios del pensamiento computacional, que no se utilizan habitualmente en la resolución de problemas, pero que, según la idea original del pensamiento computacional, estos procesos facilitarían la resolución de problemas

complejos (Wing, 2006). Estos procesos son característicos de los científicos de la computación y son la descomposición del problema como factor clave, y la automatización, el paralelismo y la simulación como habilidades complementarias que facilitan el proceso de abordar los subproblemas que hemos descompuesto. Estos procesos han sido trabajados en el curso de programación y pensamiento computacional que se ha creado para desarrollar dicho pensamiento, y poder distinguir así los dos grupos de estudio.

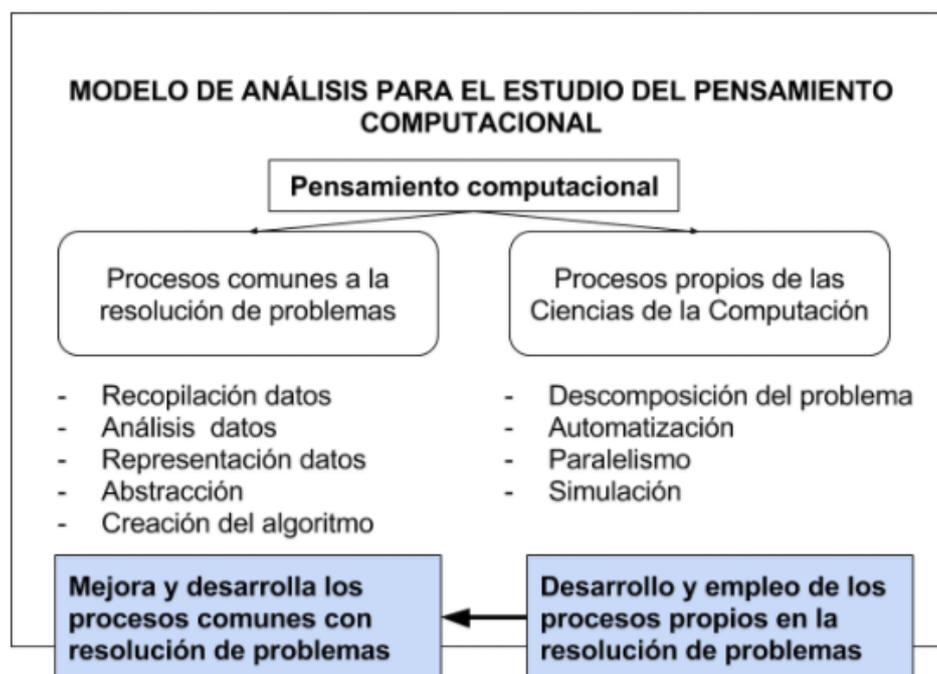


Figura 51. Modelo de análisis para el estudio del pensamiento computacional y de la estructura. Fuente: elaboración propia.

Con este marco de estudio (figura 51) hemos podido estudiar el pensamiento computacional de tal forma que, al practicar y desarrollar los procesos propios del pensamiento computacional, podamos comprobar si mejoran los procesos comunes y facilitaría la resolución de problemas complejos. Es decir, para evaluar que el pensamiento computacional facilita la RPC, tenemos que enseñar los procesos propios del PC y comprobar si han mejorado los procesos compartidos con la RPC y si se consigue un mayor éxito de resolución en las diferentes fases que compone la RP.

Además, recordemos cómo este marco de estudio está en consonancia con propuestas anteriores a la aparición del pensamiento computacional, en las que ya se reclamaban varias categorías en la base del conocimiento de las Ciencias de la Computación. Como el FITness report (Snyder, Aho, Linn, Packer, Tucker, Ullman, y Van Dam, 1999) en la que se diferenciaba entre capacidades cognitivas/intelectuales, que podríamos comparar con los procesos compartidos entre PC y RP; y conceptos computacionales y habilidades para la TIC, que podríamos comparar con los procesos propios del PC. Otro ejemplo es la propuesta realizada por Pearson y Young (2002) en el *Technically Speaking Why All Americans Need to Know More About Technology*, cuyas dimensiones se dividen en caminos de pensamiento, por una parte, y conocimiento y capacidades técnicas básicas de alfabetización tecnológica, por otra parte. Estas propuestas confirman que los procesos incluidos en el pensamiento computacional pueden abarcar diferentes dimensiones, por lo que consideramos que abordar de esta forma el PC es adecuada.

Al tratarse de un campo de investigación nuevo y prácticamente inexplorado en el área de la psicología, se ha querido partir una serie de hipótesis que nos facilitasen la exploración de los datos recogidos sobre la relación del pensamiento computacional y la resolución de problemas complejos. Estas hipótesis, recordemos, son:

1. El pensamiento computacional facilita la resolución de problemas complejos.
 - 1.1. El pensamiento computacional es especialmente útil para los problemas complejos de mayor nivel de dificultad.
 - 1.2. El curso de pensamiento computacional ha supuesto un desarrollo real en la capacidad de resolución de problemas complejos.

2. El pensamiento computacional facilita la adquisición de conocimiento sobre el problema, es decir, la representación del problema, que incluye la exploración y entendimiento de los elementos del problema, y la representación y formulación.

2.1. El pensamiento computacional facilita especialmente la formulación.

3. El pensamiento computacional facilita el proceso de resolución del problema, es decir, el control del problema, que incluye tanto la planificación y ejecución de planes para la resolución del problema, como la observación y reflexión sobre si los planes ejecutados están acercándonos a la solución del problema.

3.1. El pensamiento computacional facilita especialmente la planificación.

A partir de estas hipótesis hemos podido profundar más nuestro conocimiento sobre el pensamiento computacional y cómo éste nos puede ayudar en la resolución de problemas complejos., por lo que vamos a interpretar los resultados de cada una de ellas.

Interpretación de las hipótesis de estudio

Los resultados de nuestra investigación en relación con todas y cada una de las hipótesis planteadas, han demostrado la relación existente entre ambas variables de nuestra investigación, es decir, empleando el pensamiento computacional mejora la resolución de problemas complejos, como se ha podido comprobar en nuestra primera hipótesis, tanto en la hipótesis principal, cuyos resultados han estado a punto de arrojar diferencias significativas entre grupos, como en la hipótesis 1.2 que muestra dicha mejora.

En cuanto a cómo se relacionan ambas variables, hemos visto que la relación entre ambas se da siempre que nos enfrentamos a problemas complejos o dinámicos, ya sean estos más o menos difíciles, es decir, haya mayor número de variables o mayor interacción entre las mismas. Diríamos que el pensamiento computacional facilita la resolución de cualquier problema complejo en el que la persona necesita interactuar con

el problema para obtener toda la información disponible del mismo, independientemente de si hay mayor o menor información durante la interacción.

Por otra parte, hemos confirmado que el curso de pensamiento computacional ha desarrollado efectivamente el pensamiento computacional, ya que en el grupo experimental se han encontrado diferencias significativas entre el pretest y el postest, y en el grupo control no se han encontrado diferencias significativas, lo que quiere decir que realmente esa mejora se debe a nuestro curso de pensamiento computacional, y que dicho curso es efectivo para mejorar la forma en la que se enfrentan a problemas complejos. Con esto podemos asegurar una mayor fiabilidad en cuanto al resto de resultados de las hipótesis que nos hemos planteado, ya que podremos confirmar que, si existen diferencias, se deberán al mayor dominio del pensamiento computacional.

En cuanto a las fases de resolución de problemas en las que el pensamiento computacional aporta un mayor valor, hemos visto que en todos los análisis descriptivos se muestra que mientras que en el pretest los valores de ambos grupos son similares, en el postest están mucho más diferenciados, teniendo siempre mayor puntuación el grupo experimental que ha recibido la formación de pensamiento computacional.

En cuanto al análisis inferencial, los valores obtenidos en la T de Student no llegan a mostrar diferencias significativas, pero sí que muestran una fuerte disminución del P-Valor en el postest respecto al pretest (ver tabla 34).

Mientras que en el pretest este P-Valor era muy cercano a 1 (casi todos ellos están en 0,9), en el postest están más próximos a 0. En el postest es dónde más diferencia se ha dado entre unas hipótesis y otras, lo que nos lleva a estudiar la relación entre pensamiento computacional y resolución de problemas complejos en función de si estos valores están más próximos a 0, entendiendo que cuanta mayor proximidad exista, la relación entre ambas variables será mayor dado que habrá mayores diferencias entre los grupos.

Tabla 34. Resumen de los P-Valor obtenidos para las diferentes hipótesis en relación con las fases de RPC, tanto en el pretest como en el postest

FASE DE RPC	P-VALOR PRETEST	P-VALOR POSTEST
Rpc	0,967	0,072
Adquisición del conocimiento del problema	0,997	0,123
Representación y formulación	0,994	0,306
Aplicación del conocimiento del problema	0,930	0,071
Planificación y ejecución	0,507	0,171

En función de estos valores, podemos comprobar cómo la relación general entre pensamiento computacional y resolución de problemas complejos es cierta, ya que su P-Valor ha descendido considerablemente, acercándose al nivel de significación establecido (0,072). Además, también parece clara la influencia del pensamiento computacional en la aplicación del conocimiento, dado el P-Valor obtenido (0,071). Sin embargo, no parece tan clara la influencia del pensamiento computacional en la adquisición del conocimiento, ya que el P-Valor no ha tenido un descenso tan fuerte (0,123).

Por tanto, con estos datos podemos afirmar que el pensamiento computacional facilita la resolución de problemas complejos, y especialmente tiene mayor influencia en la aplicación del conocimiento que en la representación, aunque no se descarta que también influya en la representación.

Además, los resultados sugieren que el pensamiento computacional influye de forma general en todo el proceso de aplicación del conocimiento, es decir, no sólo influye

en la planificación y ejecución, sino que también mejora la capacidad de observación y reflexión sobre el proceso de resolución del problema.

Igualmente, aunque no quede clara la influencia del pensamiento computacional en la adquisición del problema, vemos que su influencia se daría no sólo en la representación y formulación del problema, sino que también influye positivamente en la exploración y comprensión del mismo.

Estos dos últimos resultados parecen bastante obvios, si nos paramos a pensar que este pensamiento mejorará en la representación en la medida que también mejore la exploración y el entendimiento, ya que la representación será mejor o peor en función de si esta fase previa ha sido adecuada. Lo mismo ocurriría con la aplicación del conocimiento, ya que la planificación y sobre todo la ejecución del plan, tendrán mayor éxito si somos capaces de observar y reflexionar sobre si el plan es correcto.

La interpretación de todos estos resultados nos ayuda a entender cómo se relaciona el pensamiento computacional con la resolución de problemas. Podemos afirmar que existe una relación tanto en la representación como en el proceso de resolución del problema, es decir, este pensamiento facilita tanto la representación como el proceso de resolución de problemas complejos.

El hecho de que no salgan diferencias significativas entre los grupos en ninguna de las hipótesis, no significa que trabajar este pensamiento no haya mejorado la resolución de problemas complejos. La revisión de todos los P-Valor en el pretest y el posttest de cada una de las variables analizadas, nos permite observar que sí se ha producido esta mejora, aunque no podamos considerarla una mejora significativa.

En este punto es importante señalar que se requiere mucha experiencia en un campo para desarrollar un pensamiento complejo como es el pensamiento computacional,

es decir, ser capaz de hacer pequeños cálculos de algo más extenso y complejo, por lo que la diferencia significativa entre grupos se daría en niveles superiores de experiencia.

Con esto, vemos que en nuestra investigación no nos hemos centrado en conocer si expertos en computación que tengan un desarrollo superior del pensamiento computacional mejoran en la resolución de problemas complejos. Lo que nos interesa conocer en esta investigación es cómo se desarrolla el pensamiento computacional en niveles inferiores, de cara a comprobar en qué fases de la resolución de problemas aporta mayor valor, es decir, si alguien tiene que desarrollar el pensamiento computacional, en qué aspectos de la resolución de problemas complejos debe empezar a emplear primero una estrategia computacional (en niveles inferiores), al resultar más sencillo, y en qué aspectos se requiere un mayor desarrollo de este pensamiento.

La revisión de los P-Valor, como ya hemos visto antes, nos ha ofrecido información sobre cómo se relaciona. En concreto, las diferencias encontradas entre el pretest y el posttest, indican que el pensamiento computacional facilita tanto la representación como el proceso de resolución, con unas diferencias entre el pretest y el posttest de 0,874 y 0,859, respectivamente.

Por otra parte, el hecho de que el P-Valor sea más bajo, y por tanto, esté más cerca del nivel de significación elegido, en el proceso de resolución (aplicación del conocimiento) respecto a la representación (adquisición del conocimiento) indican que en niveles iniciales del desarrollo del pensamiento computacional, es más fácil desarrollar una estrategia computacional para facilitar el proceso de resolución que para facilitar la representación, lo que nos da algunas claves para su enseñanza, como veremos más adelante.

Limitaciones de la investigación

Esta investigación se ha visto claramente limitada dado el escaso periodo de intervención realizado, lo que creemos que ha influido en el hecho de no haber obtenido diferencias significativas entre ambos grupos. Esta limitación se ha dado por la dificultad de conseguir una muestra a la que no le surgiese cualquier dificultad o problema durante la intervención, lo que nos ha supuesto abandonar dos intervenciones por no tener un suficiente control de la misma.

Finalmente, el tiempo de intervención en este tercer intento, ha sido el máximo posible, ya que después de la intervención, todos los alumnos iban a aprender programación en la asignatura de tecnología, lo que hubiese podido suponer un desarrollo en su pensamiento computacional y el no tener un grupo control real.

Otra limitación del estudio ha sido el tener que descartar tres de las preguntas de las pruebas creadas para la investigación, ya que este hecho ha afectado a poder establecer mayores diferencias gracias a una mayor variedad de pruebas y por tanto de resultados. Creemos la pregunta Climatizador 2 hubiese demostrado unas mayores diferencias al haber contado con más variabilidad en los resultados.

Nos gustaría mencionar que antes de proceder a la eliminación de los tres ítems, se realizaron unos análisis de prueba en los que sí se obtuvieron diferencias en RP, como se puede ver en la tabla 35, y en la fase de planificación y ejecución (tabla 36).

Tabla 35. Prueba T de Student para la variable resolución de problemas general, antes de eliminar las preguntas que aumentaban nuestra consistencia interna.

		Prueba de muestras independientes								
		Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias					95% de intervalo de confianza de la diferencia	
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	Inferior	Superior
PRE_Retotech	Se asumen varianzas iguales	,609	,440	-,131	36	,897	-,16246	1,24130	-2,67993	2,35500
	No se asumen varianzas iguales			-,131	34,465	,897	-,16246	1,24039	-2,68200	2,35707
POS_Retotech	Se asumen varianzas iguales	1,459	,235	2,383	36	,023	3,98319	1,67179	,59264	7,37374
	No se asumen varianzas iguales			2,443	35,986	,020	3,98319	1,63020	,67694	7,28945

Tabla 36. Prueba T de Student para la variable planificación en la resolución de problemas complejos, antes de eliminar las preguntas que aumentaban nuestra consistencia interna.

Prueba de muestras independientes										
		Prueba de Levene de calidad de varianzas		prueba t para la igualdad de medias						
		F	Sig.	t	gl	Sig. (bilateral)	Diferencia de medias	Diferencia de error estándar	95% de intervalo de confianza de la diferencia	
									Inferior	Superior
RP_Planif_PRE	Se asumen varianzas iguales	1,219	,277	,630	37	,533	,25397	,40337	-,56334	1,07127
	No se asumen varianzas iguales			,642	36,682	,525	,25397	,39556	-,54775	1,05569
RP_Planif_POS	Se asumen varianzas iguales	,004	,949	2,264	37	,030	,81746	,36107	,08586	1,54906
	No se asumen varianzas iguales			2,238	33,873	,032	,81746	,36530	,07498	1,55994

Conclusiones generales: discusión de las preguntas fundamentales de investigación

Como conclusiones generales, extraídas a partir de esta investigación, hemos podido avanzar en cuatro de las cinco cuestiones que se propusieron en el workshop sobre la naturaleza y el alcance del pensamiento computacional (NRC, 2010), descartando la quinta pregunta que tiene que ver con temas institucionales:

1. ¿Cuál es la estructura del pensamiento computacional?

El pensamiento computacional está compuesto de una serie de procesos, que se pueden dividir en dos categorías. Por una parte, encontramos procesos propios de los científicos de la computación, que facilitan la resolución de problemas complejos. Estos procesos característicos de las Ciencias de la Computación son la descomposición o modularización, la automatización, el paralelismo y la simulación. Por otra parte, encontramos procesos cognitivos necesarios para la resolución de cualquier problema, que mejoran al emplear los procesos propios de las CC. Esta mejora reclama la inclusión de los procesos como parte del pensamiento computacional, ya que la mejora se da al emplear una estrategia computacional, al ser capaz de hacer cálculos o módulos del gran problema principal.

En cuanto a cómo el pensamiento computacional mejora la resolución de problemas, es decir, en qué fases de la resolución de problemas resulta ventajoso su empleo, podemos afirmar que facilita tanto en la representación y formulación del

problema como en el proceso de resolución del mismo, aunque hay que destacar que este pensamiento ayuda especialmente al proceso de resolución del problema.

Esto podría deberse a que nuestra investigación evalúa la capacidad de resolución de problemas en personas que se encuentran en niveles inferiores de desarrollo del PC, lo que también podría significar que este pensamiento ayuda a la representación cuando la persona se encuentra en niveles superiores de desarrollo de este pensamiento.

2. ¿Cómo se puede reconocer a alguien que piensa computacionalmente?

Se podría decir que la clave del pensamiento computacional reside en la descomposición del problema para mejorar la representación, mejorando la abstracción y tratamiento de datos; y para mejorar el proceso, facilitando el proceso al crear diferentes algoritmos que puedan ser resueltos de forma independiente con diferentes técnicas como la automatización, paralelismo y simulación.

Si alguien es capaz de descomponer adecuadamente el problema, esto sería organizando la información en chunks adecuados, es decir, en grupo de información independientes que se relacionan entorno a un significado (Rips, Smith y Medin, 2012), se puede predecir que dicha persona piensa de forma computacional.

3. ¿Cuál es la conexión entre tecnología y pensamiento computacional?

En nuestra opinión se puede desarrollar un pensamiento computacional sin necesidad de estar en contacto con la tecnología, ya que este pensamiento recoge unos procesos que se pueden ver empleados por personas en contextos ajenos a la tecnología. Sin embargo, también opinamos que la tecnología es un recurso ideal para desarrollar rápidamente este pensamiento.

En cuanto a nuestra visión de cómo la tecnología se relaciona con el pensamiento computacional, entre las visiones que surgen sobre este tema (NRC, 2010), creemos que

conocer las tecnologías existentes para obtener mayores *affordances* del entorno sería suficiente para potenciar el uso del pensamiento computacional. Pero también opinamos que el aprendizaje de la programación podría facilitar en mayor medida el desarrollo de estos procesos, debido a que en programación es más habitual su empleo y por tanto la necesidad de aprender a utilizarlos correctamente.

Por tanto, nuestra primera aproximación sobre esta cuestión es que la tecnología puede ser un camino para desarrollar el PC, aunque hemos visto casos en los que no es necesaria. Por tanto, podría ser un apoyo, pero no ser fundamental.

También sería beneficiosa de cara a contextualizar el desarrollo del PC, en un primer momento para saber las *affordances* que permiten abordar mejor los problemas (conocimiento de las TIC) y en un segundo momento creando tecnología que nos permita facilitar la RP (saber programación, que conlleva a más *affordance*) y facilita el aprender a emplear cuando es necesario los procesos propios del pensamiento computacional.

4. ¿Cuál es la mejor pedagogía para promover el pensamiento computacional?

Durante esta investigación, se ha avanzado en proponer una pedagogía útil para el desarrollo de este pensamiento que consiste en la enseñanza de la programación, enfatizando en las habilidades del pensamiento computacional. Para ello, se plantean retos de programación, retos que supongan un problema a resolver por parte de los estudiantes, y se explicitan los pasos clave para trabajar la resolución de un problema utilizando estrategias propias del pensamiento computacional.

Es decir, se propone el uso de la descomposición del problema, y el uso de automatización, paralelismo y simulación, enfocadas para mejorar el tratamiento de datos, la abstracción del problema y la creación de los algoritmos que permitirán resolverlo. Además, hemos comprobado que en un primer momento de desarrollo de este

pensamiento, debemos enfocarnos a utilizar una estrategia computacional para afrontar el proceso de resolución de problemas, y posteriormente, cuando el desarrollo se encuentre en un nivel superior, abordar con una estrategia computacional la representación del problema.

La utilidad real de esta pedagogía para el desarrollo del pensamiento computacional ha quedado demostrada con treinta horas de intervención, como se ha visto en la hipótesis 1.2. Sin embargo, no podemos responder a la pregunta sobre cuál es la mejor pedagogía.

Futuras líneas de investigación

Esta investigación, como hemos visto, nos ha permitido conocer mejor la naturaleza y el alcance del pensamiento computacional, y avanzar hacia su comprensión total. Sin embargo, esta investigación ha dado lugar a otras preguntas, que necesitan ser contestadas poco a poco para poder continuar investigando sobre el tema hasta alcanzar esta comprensión total.

Investigaciones relacionadas con la representación.

En cuanto a la representación del problema, la literatura y esta investigación apuntan hacia una mejora de ésta cuando se emplea el pensamiento computacional. Por una parte, la teoría de los modelos mentales incluye la modularización como parte fundamental de la creación de modelos (Johnson-Laird, 2012), mejorando la creación de estos modelos cuanto mayor es la experiencia en modularizar los problemas. Por otra parte, tenemos la propuesta de Kwisthout (2012) que defiende que la representación se vería favorecida si utilizamos un modelo computacional.

Como futura línea de investigación, por tanto, se debería profundizar en el conocimiento sobre la influencia del pensamiento computacional en la representación.

Para ello, partiríamos del supuesto de que la representación del problema requiere un nivel superior dentro del desarrollo del pensamiento computacional.

Esta hipótesis debería comprobarse haciendo un estudio longitudinal de estudiantes que continúen estudiando programación hasta un nivel más avanzado en el que necesiten crear sistemas complejos informáticos. En este caso, se harían mediciones en varios momentos, de forma que podamos comprobar si en un principio sólo obtenemos influencia en la aplicación del conocimiento o control del problema, y posteriormente obtenemos influencia también en la adquisición del conocimiento sobre el problema.

Otro aspecto en el que se debería indagar respecto a la representación, es que necesitamos conocer si el empleo del PC mejora la descomposición del problema en partes con significado, es decir, si facilita la creación de *chunks*, o estos no tienen por qué ser unidades de conocimiento útiles como las que crean los expertos en un tema.

Es decir, hasta qué punto los factores externos e independientes de dominio, como puede ser las estrategias que emplea el pensamiento computacional, facilita la elaboración de *chunks* adecuados, o dependen en mayor medida de la experiencia de dominio.

Con el estudio de la creación de unidades de información con significado, podríamos avanzar aún más en la segunda pregunta planteada: cómo se puede reconocer a una persona que piensa computacionalmente. Teóricamente, la habilidad de descomponer el problema para facilitar el tratamiento de datos y la abstracción, mejoraría la representación del problema.

Investigaciones relacionadas con las estrategias de resolución en RPC.

Gracias a las pruebas que han sido creadas y validadas en esta investigación, se puede investigar más sobre las estrategias que son especialmente útiles para la RPC, de forma que nos permita centrarnos en la enseñanza de estas estrategias. Este asunto ya fue

demandado por Fischer, Greiff y Funke (2012), quienes exponían que esta cuestión sobre la RPC aún demanda una respuesta.

En esta línea de investigación, se podría investigar sobre cómo distintas estrategias consideradas adecuadas para la resolución de problemas complejos reducen la carga cognitiva de la persona, como ya hicieron Park, Song y Kim (2015) con el pensamiento computacional; además, por supuesto, de tener un mayor éxito en la resolución del problema a resolver.

Se necesitaría la correlación de ambas variables para considerar que una estrategia es mejor que otra en la RPC, ya que también se podría dar el caso de que una estrategia reduzca la carga cognitiva, pero no consiga mayor éxito que otras estrategias en la resolución del problema, con lo que no podríamos considerar que esa estrategia sea mejor para tal propósito.

Otras posibles investigaciones.

Queda aún mucho por conocer acerca del pensamiento computacional. Entre algunas de las cuestiones que también se requieren una investigación concreta para conocer cómo se relaciona este pensamiento con nuestro día a día, demostrando en este caso la importancia que recae en dicho pensamiento, podemos destacar dos:

La primera es en qué medida la tecnología facilita el desarrollo del pensamiento computacional de forma que nos sea útil al emplearlo para la RPC, en relación a la tercera pregunta del workshop (NRC, 2010). Esta pregunta de investigación, nos permitiría conocer qué tecnologías requieren emplear un pensamiento computacional y cómo lo requieren, en qué medida la tecnología nos puede ayudar a desarrollar este pensamiento, la correlación que podemos esperar de que un experto en tecnología vaya a emplear el

pensamiento computacional en otros contextos ajenos a la tecnología, como en tareas de su vida cotidiana.

La segunda es la relación de este pensamiento con la investigación científica, ya que en la literatura encontramos referencias a que el pensamiento computacional facilitaría la utilización de un método riguroso, necesario para llevar a cabo una investigación fiable en cualquier área de conocimiento (NRC, 2010).

Impacto de esta investigación

Impacto en el campo del pensamiento computacional

La proposición en 2006 del pensamiento computacional está propiciando toda una corriente nueva de investigación sobre cómo determinadas habilidades de pensamiento utilizadas por los científicos de la computación, es decir, los expertos en programación, puede facilitar la resolución de problemas complejos.

Por otra parte, vemos cómo en algunos lugares se está incluyendo la programación como parte del currículo educativo, como es el caso de la Comunidad de Madrid en el curso 2015/2016 con la asignatura Tecnología, Programación y Robótica en la Educación Secundaria Obligatoria.

Estas iniciativas educativas, se respaldan en la mayoría de ocasiones en que la programación es un buen recurso para desarrollar el pensamiento computacional, de hecho, en la Comunidad de Madrid se ha propuesto como un eje central de esta nueva asignatura. Con este pensamiento se desarrollaría la competencia de resolución de problemas, y por ello, aunque no todas las personas estén interesadas en aprender programación para su futuro profesional, el aprendizaje de la programación sería un recurso excelente para que cualquier persona pueda resolver problemas complejos de una forma más sencilla.

Nuestra investigación, ha confirmado que, efectivamente, enseñando programación a través de las habilidades computacionales definidas en esta misma investigación, se puede mejorar en la competencia de resolución de problemas complejos.

Además, gracias a esta investigación, hemos comprobado cómo abordar de forma exitosa el aprendizaje de la programación para el desarrollo del pensamiento computacional.

En un primer momento de enseñanza inicial de la programación, debemos trabajar un problema o reto de programación, centrándonos en el proceso de resolución del problema, el cual debemos descomponer en pequeñas partes que sean más fácilmente abordables. Además, debemos incidir en estrategias de automatización, de realización en paralelo y de simulación, para que las utilicen siempre que sea conveniente en cada una de las pequeñas partes que han podido descomponer. Para centrarnos en esta segunda fase de la resolución de problemas, el proceso de resolución, debemos proporcionar a los alumnos todos los datos relevantes y la relación entre todos estos datos, de forma que aseguremos una representación adecuada y podamos centrarnos en el proceso.

Posteriormente, cuando dominen esta primera fase de la enseñanza de la programación, en la que son capaces de emplear siempre que sea necesario diversas estrategias para solucionar un problema, pasamos a utilizar estas estrategias computacionales para conseguir una adecuada representación de un problema complejo. En un primer momento, podemos ofrecer como datos del problema todas las variables implicadas, de forma que ellos sólo tengan que explorar la relación existente entre todas ellas. En un segundo momento, cuando dominen el establecimiento de las relaciones entre variables, podemos ofrecer únicamente el enunciado del problema, de forma que ellos tengan que abordar toda la representación y ser capaces de detectar toda la información,

y dividiéndola en pequeños cálculos, en *chunks* de información con significado, que permitan representar adecuadamente el problema.

Todo este conocimiento sobre cómo enseñar programación de forma que aseguremos el desarrollo del pensamiento computacional es, sin duda, un conocimiento fundamental para la inclusión de asignaturas de programación en la educación formal, ya que nuestra misión ha sido buscar la forma en la que la enseñanza de la programación resultase beneficiosa para todos los alumnos que se viesen obligados a afrontar este aprendizaje, aunque no estuvieran interesados en utilizarlo en su vida profesional.

Impacto en el campo de resolución de problemas complejos

Esta investigación no sólo tiene repercusión en el campo de estudio del pensamiento computacional y sus implicaciones para la resolución de problemas, y para la enseñanza y desarrollo de este pensamiento, sino que tiene relevancia para el campo de investigación sobre la resolución de problemas complejos.

Esta relevancia surge del instrumento de evaluación que se ha creado para medir la resolución de problemas complejos de una forma fiable y coherente estadísticamente, un instrumento enfocado a medir la influencia de cualquier variable en esta competencia.

Como hemos visto a lo largo de la investigación, y más en profundidad en el capítulo 6 de validación de las pruebas, el instrumento creado se ha basado en un marco sólido, que ya contaba con amplios estudios psicométricos, de cara a garantizar que la medición de esta competencia se está realizando adecuadamente.

Este marco es el construido por diversos autores como Funke, Frensch, Greiff, Herde y Wüstenberg, a través de numerosas publicaciones a lo largo de estos años, que empieza con una definición de lo que consideran un problema complejo y continúa con la creación de marcos formales para evaluar de forma fiable este constructo.

Para recapitular la importancia de este marco, recordemos que el enfoque de sistemas múltiples complejos, se basa en este marco, y propone el uso de múltiples pequeñas pruebas de evaluación para evitar el efecto del contexto, al reducir errores por tratar múltiples temas. Además, estas pruebas deben ser informatizadas, de forma que se permita la interacción entre el usuario y las variables importantes del problema. Esta última característica, la interacción, es un aspecto central de los problemas complejos.

Estos autores fueron los responsables de crear el marco de pruebas de PISA 2012, que dividía la resolución de problemas en cuatro fases, a diferencia de los primeros marcos formales como MicroDYN y MicroFIN que sólo evaluaban dos fases. Además, en el marco PISA 2012, se proponen 6 niveles de dificultad para los problemas complejos, de forma que se pudiesen estimar mejor las diferencias a la hora de evaluar la resolución de problemas complejos.

Las pruebas de nuestro instrumento se basan en el marco PISA 2012. Además, nuestro instrumento incluye dos de las pruebas utilizadas en la propia prueba PISA, lo que nos ha permitido asegurar la fiabilidad de las pruebas, al estudiar la consistencia interna de las pruebas creadas con las pruebas extraídas de PISA 2012.

Por tanto, nuestra investigación tiene un gran impacto en la evaluación de la competencia de resolución de problemas complejos en entornos escolares, ya que el instrumento creado es adecuado para medir este constructo.

Referencias

CAPÍTULO 0

- CSTA y ISTE (2011) *Computational Thinking Leadership Toolkit, First edition*. Computer Science Teachers Association (CSTA) y International Society for Technology in Education (ISTE). Recuperado 13 de enero de 2015 de <http://www.iste.org/docs/ct-documents/ct-leadershipt-toolkit.pdf?sfvrsn=4>
- Herde, C. N., Wüstenberg, S., y Greiff, S. (2016). Assessment of complex problem solving: What we know and what we don't know. *Applied Measurement in Education*, 29(4), 265-277.
- Holyoak, K. J. y Morrison, R. G. (2012) Thinking and reasoning: A reader's guide. En Holyoak, K. J. y Morrison, R. G. (Eds.) *The oxford handbook of thinking and reasoning*, 1-9. Oxford University Press.
- Kwisthout, J. (2012). Relevancy in problem solving: a computational framework. *The Journal of Problem Solving*, 5(1), 4.
- National Research Council (2010) *Report of a Workshop on the Scope and Nature of Computational Thinking*. Washington, DC: The National Academies Press
- Valverde-Berrocoso, J.; Fernández-Sánchez, M. y Garrido-Arroyo, M. (2015) El pensamiento computacional y las nuevas ecologías del aprendizaje. *Revista de Educación a Distancia*. (46)
- Wing, J. M. (2006) Computational thinking. *Communications of the ACM*, 49, 33-35
- Wing, J. M. (2008) Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 366, 3717-3725. The Royal Society
- Wing, J. M. (2011) Research Notebook: Computational Thinking: What and Why? *The Link*. Pittsburgh, PA: Carneige Mellon. Recuperado de <https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>
- Wing, J. M. (2014). Computational thinking benefits society. *Social Issues in Computing*. New York: Academic Press.

Zapata-Ros, M. (2015) Pensamiento computacional: Una nueva alfabetización digital
Revista de Educación a Distancia. Universidad de Murcia

CAPÍTULO 1

Brennan, K. y Resnick, M. (2012) New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 annual meeting of the American Educational Research Association*. Vancouver, Canada

Carretero, M. y Asensio, M. (2004) Introducción. *Psicología del Pensamiento*, 13-34. Madrid: Alianza

CSTA y ISTE (2011) *Computational Thinking Leadership Toolkit, First edition*. Computer Science Teachers Association (CSTA) y International Society for Technology in Education (ISTE). Recuperado 13 de enero de 2015 de <http://www.iste.org/docs/ct-documents/ct-leadership-toolkit.pdf?sfvrsn=4>

Cuny, J.; Snyder, L. y Wing, J. M. (2010) Demystifying computational thinking for non-computer scientists Unpublished manuscript in progress, referenced in <http://www.cs.cmu.edu/%7ECompThink/resources/TheLinkWing.pdf>

Greiff, S., Wüstenberg, S., y Funke, J. (2012). Dynamic problem solving: A new assessment perspective. *Applied Psychological Measurement*, 36(3), 189-213.

Hartmanis, J., y Lin, H. (1992). *Computing the future: a broader agenda for computer science and engineering*. National Academy Press.

Holyoak, K. J. y Morrison, R. G. (2012) Thinking and reasoning: A reader's guide. En Holyoak, K. J. y Morrison, R. G. (Eds.) *The oxford handbook of thinking and reasoning*, 1-9. Oxford University Press.

Hu, C. (2011) Computational thinking: what it might mean and what we might do about it *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, 223-227

National Research Council (2010) *Report of a Workshop on the Scope and Nature of Computational Thinking*. Washington, DC: The National Academies Press

- National Research Council (2011) *Report of a workshop of pedagogical aspects of computational thinking*. Washington, DC: The National Academies Press
- OECD (2010). *PISA 2012 field trial: problem solving framework*. Paris: OECD
- Papert, S. (1972). Teaching children thinking. *Programmed Learning and Educational Technology*, 9(5), 245-255
- Pearson, G., y Young, A. T. (Eds.). (2002). *Technically speaking: Why all Americans need to know more about technology*. National Academies Press.
- Selby, C. y Woollard, J. (2013) *Computational thinking: the developing definition* University of Southampton (E-prints), University of Southampton (E-prints)
- Shaw, M. (2004). *Computer science: Reflections on the field, reflections from the field*. National Academy Press
- Snyder, L., Aho, A. V., Linn, M., Packer, A., Tucker, A., Ullman, J., y Van Dam, A. (1999). *Being fluent with information technology*. Computer Science and Telecommunications Board, National Research Council. National Academy Press, Washington, DC.
- Teacher Advisory Council (2009). *Engineering in K-12 education: Understanding the status and improving the prospects*. National Academies Press.
- Valverde-Berrocoso, J.; Fernández-Sánchez, M. y Garrido-Arroyo, M. (2015) El pensamiento computacional y las nuevas ecologías del aprendizaje. *Revista de Educación a Distancia*. (46)
- Villatoro, F. R. (16 de Diciembre de 2012). *Nota dominical: Richard Feynman, los ordenadores y los métodos numéricos* [Mensaje en un blog]. Recuperado de <http://francis.naukas.com/2012/12/16/nota-dominical-richard-feynman-los-ordenadores-y-los-metodos-numericos/>
- Weinberg, A. E. (2013) *Computational thinking: An investigation of the existing scholarship and research*. Doctoral dissertation, Colorado State University. Libraries.
- Wing, J.M. (2006) Computational thinking. *Communications of the ACM*, 49, 33-35

- Wing, J. M. (2008) Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 366, 3717-3725. The Royal Society
- Wing, J. M. (2011) Research Notebook: Computational Thinking: What and Why? *The Link*. Pittsburgh, PA: Carneige Mellon. Recuperado de <https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>
- Wing, J. M. (2014). Computational thinking benefits society. *Social Issues in Computing*. New York: Academic Press.
- Zapata-Ros, M. (2015) Pensamiento computacional: Una nueva alfabetización digital *Revista de Educación a Distancia*. Universidad de Murcia

CAPÍTULO 2

- Ambrosio, A. P., Almeida, L. S., Macedo, J., y Franco, A. H. R. (2014). Exploring core cognitive skills of Computational Thinking. In *Psychology of Programming Interest Group Annual Conference 2014 (PPIG 2014)* (pp. 25-34).
- Baddeley, A. D. (1992). Working Memory and Conscious Awareness. *Theories of memory*, 11-20.
- Baddeley, A. D., y Hitch, G. (1974). Working memory. *Psychology of learning and motivation*, 8, 47-89.
- Bassok, M., y Novick, L. R. (2012). Problem solving. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, (p. 413-432). Oxford University Press.
- Birch, H. G., y Rabinowitz, H. S. (1951). The negative effect of previous experience on productive thinking. *Journal of experimental psychology*, 41(2), 121.
- Bowden, E. M., Jung-Beeman, M., Fleck, J., y Kounios, J. (2005). New approaches to demystifying insight. *Trends in cognitive sciences*, 9(7), 322-328.
- Cheng, P.W and Buehner, M.J (2012) Causal Learning. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, 134-145. Oxford University Press.
- Chiappe, D., y MacDonald, K. (2005). The evolution of domain-general mechanisms in intelligence and learning. *The Journal of general psychology*, 132(1), 5-40.

- Chu, Y., Li, Z., Su, Y., y Pizlo, Z. (2010). Heuristics in Problem Solving: The Role of Direction in Controlling Search Space. *The Journal of Problem Solving*, 3(1), 3.
- Chuderski, A. (2014). How well can storage capacity, executive control, and fluid reasoning explain insight problem solving. *Intelligence*, 46, 258-270.
- CSTA y ISTE (2011) *Computational Thinking Leadership Toolkit, First edition*. Computer Science Teachers Association (CSTA) y International Society for Technology in Education (ISTE). Recuperado 13 de enero de 2015 de <http://www.iste.org/docs/ct-documents/ct-leadership-toolkit.pdf?sfvrsn=4>
- Duncker, K. (1945) On problem-solving (L. S. Lees, Trans.) *Psychological Monographs*, 58 (Whole No. 270). (Original work published 1935.)
- Dove, G. (2009) Beyond perceptual symbols: A call for representational pluralism. *Cognition*, 110, 412-431.
- Ericsson, K. A., y Kintsch, W. (1995). Long-term working memory. *Psychological review*, 102(2), 211.
- Estudillo, A. J., Bermudo, E., Casado, N., Prasad, J., y Garcia-Orza, J. (2015). Automaticity in subtractions depends on problem-size. *Anales de Psicología/Annals of Psychology*, 31(2), 697-704.
- Fischer, A., Greiff, S., y Funke, J. (2012). The process of solving complex problems. *The Journal of Problem Solving*, 4 (1), 19-42.
- Funke, J. (2001). Dynamic systems as tools for analysing human judgement. *Thinking y Reasoning*, 7(1), 69-89.
- Funke, J. (2013). Human problem solving in 2012. *The Journal of Problem Solving*, 6(1), 3.
- Funke, J., y Frensch, P. (1995). Complex problem solving research in North America and Europe: An integrative review. *Foreign Psychology*, 5, 42-47.
- Greiff, S., y Fischer, A. (2013). Measuring Complex Problem Solving: An educational application of psychological theories. *Journal for educational research online*, 5(1), 38.

- Greitemeyer, T., y Würz, D. (2006). Mental simulation and the achievement of health goals: The role of goal difficulty. *Imagination, Cognition and Personality*, 25(3), 239-251.
- Gobet, F., Lloyd-Kelly, M. y Lane, P.C.R. (2016). "What's in a Name? The Multiple Meanings of "Chunk" y "Chunking."". *Front. Psychol.* 7
- Hambrick, D. Z., y Engle, R. W. (2003). The role of working memory in problem solving. *The psychology of problem solving*, 176-206.
- Hegarty, M., y Stull, A. T. (2012). Visuospatial thinking. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*. Oxford University Press.
- Herde, C. N., Wüstenberg, S., y Greiff, S. (2016). Assessment of complex problem solving: What we know and what we don't know. *Applied Measurement in Education*, 29(4), 265-277.
- Hoddeson, L. (2002). Toward a history-based model for scientific invention: Problem-solving practices in the invention of the transistor and the development of the theory of superconductivity. *Mind y Society*, 3(1), 67-79.
- Holyoak, K. J. (2012) Analogy y relational reasoning. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, (p. 234-259). Oxford University Press.
- Johnson-Laird, P. N. (1980). Mental models in cognitive science. *Cognitive science*, 4(1), 71-115.
- Johnson-Laird, P. N. (2012) Inference with mental models. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, 134-145. Oxford University Press.
- Kotovsky, K., y Simon, H. A. (1990). What makes some problems really hard: Explorations in the problem space of difficulty. *Cognitive psychology*, 22(2), 143-183.
- Markman, A. B. (2012) Knowledge representation. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, (p. 36-51). Oxford University Press.
- Markman, A. B., y Dietrich, E. (2000). In defense of representation. *Cognitive Psychology*, 40(2), 138-171.7
- Mayer, R. E. (1995) The search for insight: Grappling with Gestalt psychology's unanswered questions. En Sternberg, R. J. y Davidson, J. E. (Eds.) *The nature of insight*. (pp. 3-32). Cambridge, MA, US: The MIT Press.

- Morrison, R. G. (2005) Thinking in Working Memory. En Holyoak y Morrison (Eds.) *The Cambridge handbook of thinking y reasoning*, Cap. 19 (p. 457-473). NY: Cambridge University Press.
- Newell, A., y Simon, H. A. (1972). *Human problem solving* (Vol. 104, No. 9). Englewood Cliffs, NJ: Prentice-Hall.
- Novick, L. R. y Bassok, M (2005) Problem Solving. En Holyoak y Morrison (Eds.) *The Cambridge handbook of thinking y reasoning*, (p. 321-349). NY: Cambridge University Press.
- Ohlsson, S. (2012). The problems with problem solving: Reflections on the rise, current status, and possible future of a cognitive research paradigm. *The Journal of Problem Solving*, 5(1), 7.
- Polson, P. G., y Jeffries, R. (2014). Instruction in general problem-solving skills: An analysis of four approaches. *Thinking and Learning Skills: Volume 1: Relating Instruction To Research*, 417.
- Postigo, Y. (2004) Estrategias en solución de problemas. En Carretero, M. y Asensio, M. (Eds.) *Psicología del pensamiento: Teoría y prácticas*. Madrid: Alianza
- Reitman, W. R. (1965) *Cognition and thought: an information processing approach*. New York: Wiley
- Rips, L. J., Smith, E. E. y Medin, D. L. (2012) Concepts y categories memory. Meaning y metaphysics. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, (p. 177-209). Oxford University Press.
- Rodríguez Palmero, M. L., Marrero Acosta, J., y Moreira, M. A. (2001). La Teoría de los Modelos Mentales de Johnson-Laird y sus principios: una aplicación con modelos mentales de célula en estudiantes del Curso de Orientación Universitaria. *Investigações em ensino de ciências. Porto Alegre*. Vol. 6, n. 3 (set./dez. 2001), p. 243-268.
- Sánchez Miguel, E. (2008). La comprensión lectora. En *La lectura en España. Informe 2008: leer para aprender* (pp. 191-208). Federación de Gremios de Editores de España.

- Tversky, B. (2005) Visuospatial Reasoning. En Holyoak y Morrison (Eds.) *The Cambridge handbook of thinking y reasoning*, (p. 209-240). NY: Cambridge University Press.
- van Steenburgh, J.J, Fleck, J.I, Beeman,M. y Kounios, J. (2012) Insight. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, (pp. 475-491). Oxford University Press.
- Vollmeyer, R., Burns, B. D., y Holyoak, K. J. (1996). The impact of goal specificity on strategy use and the acquisition of problem structure. *Cognitive Science*, 20, 75–100.
- Weisberg, R. W. (2006). *Creativity: Understanding innovation in problem solving, science, invention, and the arts*. John Wiley y Sons.
- Weisberg, R. W. (2015). Toward an integrated theory of insight in problem solving. *Thinking y Reasoning*, 21(1), 5-39.
- Weisberg, R. W. y Alba, J. W. (1981) Gestalt theory, insight, y past experience: Reply to Dominowski. *Journal of Experimental Psychology: General*, Vol 110(2), 199-203.
- Wertheimer, M. (1959). *Productive thinking*. New York: Harper.
- Wiley, J., y Jarosz, A. F. (2012). How Working Memory Capacity Affects Problem Solving. *Psychology of Learning y Motivation-Advances in Research y Theory*, 56, 185. ISBN: 978-0-12-394393-4

CAPÍTULO 3

- Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal*, 55(7), 832-835.
- Angeli, C., y Valanides, N. (2013). Technology mapping: An approach for developing technological pedagogical content knowledge. *Journal of Educational Computing Research*, 48(2), 199-221.
- Baddeley, A. D., y Hitch, G. (1974). Working memory. *Psychology of learning and motivation*, 8, 47-89.
- Basawapatna, A., Koh, K. H., Repenning, A., Webb, D. C., y Marshall, K. S. (2011). Recognizing computational thinking patterns. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (pp. 245-250). ACM.

- Blackwell, A. F., Church, L., y Green, T. R. G. (2008, September). The abstract is 'an enemy': Alternative perspectives to computational thinking. In *Proceedings PPIG* (Vol. 8, pp. 34-43).
- Brennan, K. y Resnick, M. (2012) New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 annual meeting of the American Educational Research Association*. Vancouver, Canada
- Carretero y Asensio (2004) Introducción. *Psicología del pensamiento*. pp. 13-34. Madrid: Alianza.
- Carruthers, S., y Stege, U. (2013). On evaluating human problem solving of computationally hard problems. *The Journal of Problem Solving*, 5(2), 4.
- CSTA y ISTE (2011) *Computational Thinking Leadership Toolkit, First edition*. Computer Science Teachers Association (CSTA) y International Society for Technology in Education (ISTE). Recuperado 13 de enero de 2015 de <http://www.iste.org/docs/ct-documents/ct-leadership-toolkit.pdf?sfvrsn=4>
- Estudillo, A. J., Bermudo, E., Casado, N., Prasad, J., y Garcia-Orza, J. (2015). Automaticity in subtractions depends on problem-size. *Anales de Psicología/Annals of Psychology*, 31(2), 697-704.
- Funke, J. (2001). Dynamic systems as tools for analysing human judgement. *Thinking y Reasoning*, 7(1), 69-89.
- Greitemeyer, T., y Würz, D. (2006). Mental simulation and the achievement of health goals: The role of goal difficulty. *Imagination, Cognition and Personality*, 25(3), 239-251.
- Hoddeson, L. (2002). Toward a history-based model for scientific invention: Problem-solving practices in the invention of the transistor and the development of the theory of superconductivity. *Mind y Society*, 3(1), 67-79.
- Ioannidou, A., Bennett, V., Reppenning, A., Koh, K. H., y Basawapatna, A. (2011). Computational Thinking Patterns. *Online Submission*.
- Johnson, A. P. (2000). *Up and out: Using creative and critical thinking skills to enhance learning*. Allyn y Bacon.

- Kroll, J. F., y Linck, J. A. (2007). Representation and skill in second language learners and proficient bilinguals. *Cognitive aspects of bilingualism*, 237-269.
- Kwisthout, J. (2012). Relevancy in problem solving: a computational framework. *The Journal of Problem Solving*, 5(1), 4.
- Markman, A. B. (2012) Knowledge representation. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, (p. 36-51). Oxford University Press.
- Matlock, T. (2004). Fictive motion as cognitive simulation. *Memory y Cognition*, 32(8), 1389-1400.
- National Research Council (2010) *Report of a Workshop on the Scope and Nature of Computational Thinking*. Washington, DC: The National Academies Press
- Novick, L. R. y Bassok, M (2005) Problem Solving. En Holyoak y Morrison (Eds.) *The Cambridge handbook of thinking y reasoning*, (p. 321-349). NY: Cambridge University Press.
- Park, S. Y., Song, K. S., y Kim, S. (2015). EEG Analysis for Computational Thinking based Education Effect on the Learners' Cognitive Load. *Proceedings of the Applied Computer and Applied Computational Science (ACACOS'15), Kuala Lumpur, Malaysia*, 23-25.
- Penner, B. C., y Voss, J. F. (1983). Problem Solving Skills in the Social Sciences: Methodological Considerations.
- Polson, P. G., y Jeffries, R. (2014). Instruction in general problem-solving skills: An analysis of four approaches. *Thinking and Learning Skills: Volume 1: Relating Instruction To Research*, 417.
- Rosales, J., Vicente, S., Chamoso, J. M., Muñoz, D., y Orrantia, J. (2012). Teacher–student interaction in joint word problem solving. The role of situational and mathematical knowledge in mainstream classrooms. *Teaching and Teacher Education*, 28(8), 1185-1195.
- Schoppek, W. (2002). Examples, rules, and strategies in the control of dynamic systems. *Cognitive Science Quarterly*, 2(1), 63-92.

- Selby, C. y Woollard, J. (2013) *Computational thinking: the developing definition University of Southampton* (E-prints), University of Southampton (E-prints)
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive science*, 12(2), 257-285.
- Weisberg, R. W. (2015). Toward an integrated theory of insight in problem solving. *Thinking y Reasoning*, 21(1), 5-39.
- Wiley, J., y Jarosz, A. F. (2012). How Working Memory Capacity Affects Problem Solving. *Psychology of Learning y Motivation-Advances in Research y Theory*, 56, 185. ISBN: 978-0-12-394393-4
- Wing, J. M. (2006) Computational thinking. *Communications of the ACM*, 49, 33-35
- Wing, J. M. (2008) Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 366, 3717-3725. The Royal Society
- Wing, J. M. (2011) Research Notebook: Computational Thinking: What and Why? *The Link*. Pittsburgh, PA: Carneige Mellon. Recuperado de <https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>

CAPÍTULO 4

- Applegate, D. L., Bixby, R. E., Chvatal, V., y Cook, W. J. (2011). *The traveling salesman problem: a computational study*. Princeton university press.
- Berg, E. A. (1948). A simple objective technique for measuring flexibility in thinking. *The Journal of general psychology*, 39(1), 15-22.
- Brehmer, B., y Dörner, D. (1993). Experiments with computer-simulated microworlds: Escaping both the narrow straits of the laboratory and the deep blue sea of the field study. *Computers in Human Behavior*, 9(2), 171-184.

- Brennan, K. y Resnick, M. (2012) New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 annual meeting of the American Educational Research Association*. Vancouver, Canada
- Chronicle, E. P., MacGregor, J. N., Lee, M., Ormerod, T. C., y Hughes, P. (2008). Individual differences in optimization problem solving: Reconciling conflicting results. *The Journal of Problem Solving*, 2(1), 4.
- CSTA y ISTE (2011) *Computational Thinking Leadership Toolkit, First edition*. Computer Science Teachers Association (CSTA) y International Society for Technology in Education (ISTE). Recuperado 13 de enero de 2015 de <http://www.iste.org/docs/ct-documents/ct-leadershiptoolkit.pdf?sfvrsn=4>
- Dörner, D. (1997). *The logic of failure: Recognizing and avoiding error in complex situations*. New York: Basic Books.
- Ercikan, K., y Oliveri, M. E. (2016). In search of validity evidence in support of the interpretation and use of assessments of complex constructs: Discussion of research on assessing 21st century skills. *Applied Measurement in Education*, 29(4), 310-318.
- Frensch, P. A., y Funke, J. (2014). *Complex problem solving: The European perspective*. Psychology Press.
- Funke, J. (2001). Dynamic systems as tools for analysing human judgement. *Thinking y Reasoning*, 7(1), 69-89.
- Funke, J. (2013). Human problem solving in 2012. *The Journal of Problem Solving*, 6(1), 3.
- Funke, J., y Frensch, P. (1995). Complex problem solving research in North America and Europe: An integrative review. *Foreign Psychology*, 5, 42-47.
- Greiff, S., y Fischer, A. (2013). Measuring Complex Problem Solving: An educational application of psychological theories. *Journal for educational research online*, 5(1), 38.

- Greiff, S., Fischer, A., Wüstenberg, S., Sonnleitner, P., Brunner, M., y Martin, R. (2013). A multitrait–multimethod study of assessment instruments for complex problem solving. *Intelligence*, 41(5), 579-596.
- Greiff, S., Holt, D. V., y Funke, J. (2013). Perspectives on problem solving in educational assessment: analytical, interactive, and collaborative problem solving. *Journal of Problem Solving*, 5(2), 71-91.
- Greiff, S., Wüstenberg, S., y Funke, J. (2012). Dynamic problem solving: A new assessment perspective. *Applied Psychological Measurement*, 36(3), 189-213.
- Herde, C. N., Wüstenberg, S., y Greiff, S. (2016). Assessment of complex problem solving: What we know and what we don't know. *Applied Measurement in Education*, 29(4), 265-277.
- Koh, K. H., Basawapatna, A., Nickerson, H., y Repenning, A. (2014, July). Real time assessment of computational thinking. In *Visual Languages and Human-Centric Computing (VL/HCC), 2014 IEEE Symposium on* (pp. 49-52). IEEE.
- Koh, K. H., Nickerson, H., Basawapatna, A., y Repenning, A. (2014, June). Early validation of computational thinking pattern analysis. In *Proceedings of the 2014 conference on Innovation y technology in computer science education* (pp. 213-218). ACM.
- Lye, S. Y., y Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51-61.
- Moreno-León, J., Robles, G., y Román-González, M. (2015). Dr. Scratch: Análisis Automático de Proyectos Scratch para Evaluar y Fomentar el Pensamiento Computacional. *Revista de Educación a Distancia*, (46).
- Mueller, S. T., Perelman, B. S., Tan, Y. Y., y Thanasuan, K. (2015). Development of the PEBL Traveling Salesman Problem Computerized Testbed. *The Journal of Problem Solving*, 8(1), 4.

- Novick, L. R. y Bassok, M (2005) Problem Solving. En Holyoak y Morrison (Eds.) *The Cambridge handbook of thinking y reasoning*, (p. 321-349). NY: Cambridge University Press.
- OECD (2013) Problem-Solving Framework. In *PISA 2012 Assessment and Analytical Framework: Mathematics, Reading, Science, Problem Solving and Financial Literacy*. OECD Publishing
- Park, S. Y., Song, K. S., y Kim, S. (2015). EEG Analysis for Computational Thinking based Education Effect on the Learners' Cognitive Load. *Proceedings of the Applied Computer and Applied Computational Science (ACACOS'15), Kuala Lumpur, Malaysia*, 23-25.
- Ritchhart, R. y Perkins, D. N. (2005) Learning to Think: The Challenges of Teaching Thinking. En Holyoak, K. J. y Morrison, R. G. (Eds.) *The Cambridge Handbook of Thinking and Reasoning*. (pp. 775-802). Cambridge University Press.
- Smith, E. E., y Kosslyn, S. M. (2008). *Procesos cognitivos: modelos y bases neurales*. 978-84-8322-396-3. Madrid: Pearson Educación
- Sternberg, R. J., y Frensch, P. A. (Eds.). (1991). *Complex problem solving: Principles and mechanisms*. Hillsdale, NJ: Lawrence Erlbaum.
- Zapata-Ros, M. (2015) Pensamiento computacional: Una nueva alfabetización digital *Revista de Educación a Distancia*. Universidad de Murcia

CAPÍTULO 5

- Greiff, S., Wüstenberg, S., y Funke, J. (2012). Dynamic problem solving: A new assessment perspective. *Applied Psychological Measurement*, 36(3), 189-213.
- Instituto Nacional de Evaluación Educativa (2013) *Pisa 2012. Resolución de problemas. Ejemplos de preguntas en soporte digital*. España: MECD
- Marais, C., y Bradshaw, K. (2015). Problem-solving ability of first year CS students: A case study and intervention. In *Proceedings of the 44th Conference of the Southern African Computers Lecturers' Association*.

OECD (2013) Problem-Solving Framework. In *PISA 2012 Assessment and Analytical Framework: Mathematics, Reading, Science, Problem Solving and Financial Literacy*. OECD Publishing

CAPÍTULO 6

Blackwell, A. F., Church, L., y Green, T. R. G. (2008) The abstract is 'an enemy': Alternative perspectives to computational thinking. In *Proceedings PPIG* Vol. 8, pp. 34-43.

Funke, J. (2001). Dynamic systems as tools for analysing human judgement. *Thinking y Reasoning*, 7(1), 69-89.

Funke, J. (2013). Human problem solving in 2012. *The Journal of Problem Solving*, 6(1), 3.

Funke, J., y Frensch, P. (1995). Complex problem solving research in North America and Europe: An integrative review. *Foreign Psychology*, 5, 42-47.

Greiff, S., Holt, D. V., y Funke, J. (2013). Perspectives on problem solving in educational assessment: analytical, interactive, and collaborative problem solving. *Journal of Problem Solving*, 5(2), 71-91.

Greiff, S., Wüstenberg, S., y Funke, J. (2012). Dynamic problem solving: A new assessment perspective. *Applied Psychological Measurement*, 36(3), 189-213.

Greiff, S., Wüstenberg, S., Molnár, G., Fischer, A., Funke, J., y Csapó, B. (2013). Complex Problem Solving in Educational Contexts—Something Beyond g: Concept, Assessment, Measurement Invariance, and Construct Validity. *Journal of Educational Psychology*, 105(2), 364.

Herde, C. N., Wüstenberg, S., y Greiff, S. (2016). Assessment of complex problem solving: What we know and what we don't know. *Applied Measurement in Education*, 29(4), 265-277.

Instituto Nacional de Evaluación Educativa (2013) *Pisa 2012. Resolución de problemas. Ejemplos de preguntas en soporte digital*. España: MECD

- Kwisthout, J. (2012). Relevancy in problem solving: a computational framework. *The Journal of Problem Solving*, 5(1), 4.
- OECD (2010) *PISA 2012 field trial problem solving framework: Draft subject to possible revision after the field trial*. Paris: OECD. Disponible en: <http://www.oecd.org/pisa/pisaproducts/46962005.pdf>.
- Schleicher (2006) Fundamentos y cuestiones políticas subyacentes al desarrollo de PISA. *Revista de educación*, 21-43
- Schoppek, W. (2002). Examples, rules, and strategies in the control of dynamic systems. *Cognitive Science Quarterly*, 2(1), 63-92.
- Wing, J. M. (2006) Computational thinking. *Communications of the ACM*, 49, 33-35
- Wing, J. M. (2014). Computational thinking benefits society. *Social Issues in Computing*. New York: Academic Press.

CAPÍTULO 8

- Blackwell, A. F., Church, L., y Green, T. R. G. (2008, September). The abstract is' an enemy': Alternative perspectives to computational thinking. In *Proceedings PPIG* (Vol. 8, pp. 34-43).
- CSTA y ISTE (2011) *Computational Thinking Leadership Toolkit, First edition*. Computer Science Teachers Association (CSTA) y International Society for Technology in Education (ISTE). Recuperado 13 de enero de 2015 de <http://www.iste.org/docs/ct-documents/ct-leadershiptoolkit.pdf?sfvrsn=4>
- Fischer, A., Greiff, S., y Funke, J. (2012). The process of solving complex problems. *The Journal of Problem Solving*, 4 (1), 19-42.
- Hu, C. (2011) Computational thinking: what it might mean and what we might do about it *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, 223-227
- Johnson-Laird, P. N. (2012) Inference with mental models. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, 134-145. Oxford University Press.

- Kwisthout, J. (2012). Relevancy in problem solving: a computational framework. *The Journal of Problem Solving*, 5(1), 4.
- National Research Council (2010) *Report of a Workshop on the Scope and Nature of Computational Thinking*. Washington, DC: The National Academies Press
- Park, S. Y., Song, K. S., y Kim, S. (2015). EEG Analysis for Computational Thinking based Education Effect on the Learners' Cognitive Load. *Proceedings of the Applied Computer and Applied Computational Science (ACACOS'15)*, Kuala Lumpur, Malaysia, 23-25.
- Pearson, G., y Young, A. T. (Eds.). (2002). *Technically speaking: Why all Americans need to know more about technology*. National Academies Press.
- Rips, L. J., Smith, E. E. y Medin, D. L. (2012) Concepts y categories memory. Meaning y metaphysics. En Holyoak y Morrison (Eds.) *The Oxford handbook of thinking y reasoning*, (p. 177-209). Oxford University Press.
- Selby, C. y Woollard, J. (2013) *Computational thinking: the developing definition University of Southampton* (E-prints), University of Southampton (E-prints)
- Snyder, L., Aho, A. V., Linn, M., Packer, A., Tucker, A., Ullman, J., y Van Dam, A. (1999). *Being fluent with information technology*. Computer Science and Telecommunications Board, National Research Council. National Academy Press, Washington, DC.
- Wing, J. M. (2006) Computational thinking. *Communications of the ACM*, 49, 33-35

ANEXOS

Anexo 1. Distribución de los aciertos en cada pregunta (muestra validación)

age1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	15	53,6	53,6	53,6
	1	11	39,3	39,3	92,9
	2	2	7,1	7,1	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

age2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	14	50,0	50,0	50,0
	1	14	50,0	50,0	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

vid1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	15	53,6	53,6	53,6
	1	13	46,4	46,4	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

vid2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	16	57,1	57,1	57,1
	1	7	25,0	25,0	82,1
	2	5	17,9	17,9	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

bill1ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	13	46,4	46,4	46,4
	1	15	53,6	53,6	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

bill2ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	18	64,3	64,3	64,3
	1	5	17,9	17,9	82,1
	2	5	17,9	17,9	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

bill3ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	15	53,6	53,6	53,6
	1	13	46,4	46,4	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

tren1ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	16	57,1	57,1	57,1
	1	12	42,9	42,9	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

tren2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	25	89,3	89,3	89,3
	1	3	10,7	10,7	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

cli1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	11	39,3	39,3	39,3
	1	17	60,7	60,7	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

cli2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	12	42,9	42,9	42,9
	1	13	46,4	46,4	89,3
	2	3	10,7	10,7	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

coche1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	19	67,9	67,9	67,9
	1	9	32,1	32,1	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

coche2ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	20	71,4	71,4	71,4
	1	7	25,0	25,0	96,4
	2	1	3,6	3,6	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

coche3ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	11	39,3	39,3	39,3
	1	16	57,1	57,1	96,4
	2	1	3,6	3,6	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

coche4ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	26	92,9	92,9	92,9
	1	2	7,1	7,1	100,0
	Total	28	100,0	100,0	

a. Grupo = 1

Anexo 2. Distribución de los aciertos en cada pregunta (muestra pretest)

age1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	23	60,5	60,5	60,5
	1	14	36,8	36,8	97,4
	2	1	2,6	2,6	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

age2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	11	28,9	28,9	28,9
	1	27	71,1	71,1	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

vid1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	23	60,5	60,5	60,5
	1	15	39,5	39,5	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

vid2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	18	47,4	47,4	47,4
	1	19	50,0	50,0	97,4
	2	1	2,6	2,6	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

bill1ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	11	28,9	28,9	28,9
	1	27	71,1	71,1	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

bill2ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	21	55,3	55,3	55,3
	1	10	26,3	26,3	81,6
	2	7	18,4	18,4	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

bill3ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	15	39,5	39,5	39,5
	1	23	60,5	60,5	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

tren1ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	13	34,2	34,2	34,2
	1	25	65,8	65,8	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

tren2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	30	78,9	78,9	78,9
	1	5	13,2	13,2	92,1
	2	3	7,9	7,9	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

cli1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	16	42,1	42,1	42,1
	1	22	57,9	57,9	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

cli2^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	15	39,5	39,5	39,5
	1	17	44,7	44,7	84,2
	2	6	15,8	15,8	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

coche1^a

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	25	65,8	65,8	65,8
	1	9	23,7	23,7	89,5
	2	4	10,5	10,5	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

coche2ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	22	57,9	57,9	57,9
	1	12	31,6	31,6	89,5
	2	4	10,5	10,5	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

coche3ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	6	15,8	15,8	15,8
	1	31	81,6	81,6	97,4
	2	1	2,6	2,6	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

coche4ª

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	34	89,5	89,5	89,5
	1	4	10,5	10,5	100,0
	Total	38	100,0	100,0	

a. Grupo = 2

Anexo 3. Distribución de los aciertos en cada pregunta (ambas muestras)

age1

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	38	57,6	57,6	57,6
	1	25	37,9	37,9	95,5
	2	3	4,5	4,5	100,0
	Total	66	100,0	100,0	

age2

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	25	37,9	37,9	37,9
	1	41	62,1	62,1	100,0
	Total	66	100,0	100,0	

vid1

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	38	57,6	57,6	57,6
	1	28	42,4	42,4	100,0
	Total	66	100,0	100,0	

vid2

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	34	51,5	51,5	51,5
	1	26	39,4	39,4	90,9
	2	6	9,1	9,1	100,0
	Total	66	100,0	100,0	

bill1

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	24	36,4	36,4	36,4
	1	42	63,6	63,6	100,0
	Total	66	100,0	100,0	

bill2

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	39	59,1	59,1	59,1
	1	15	22,7	22,7	81,8
	2	12	18,2	18,2	100,0
	Total	66	100,0	100,0	

bill3

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	30	45,5	45,5	45,5
	1	36	54,5	54,5	100,0
	Total	66	100,0	100,0	

tren1

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	29	43,9	43,9	43,9
	1	37	56,1	56,1	100,0
	Total	66	100,0	100,0	

tren2

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	55	83,3	83,3	83,3
	1	8	12,1	12,1	95,5
	2	3	4,5	4,5	100,0
	Total	66	100,0	100,0	

cli1

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	27	40,9	40,9	40,9
	1	39	59,1	59,1	100,0
	Total	66	100,0	100,0	

cli2

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	27	40,9	40,9	40,9
	1	30	45,5	45,5	86,4
	2	9	13,6	13,6	100,0
	Total	66	100,0	100,0	

coche1

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	44	66,7	66,7	66,7
	1	18	27,3	27,3	93,9
	2	4	6,1	6,1	100,0
	Total	66	100,0	100,0	

coche2

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	42	63,6	63,6	63,6
	1	19	28,8	28,8	92,4
	2	5	7,6	7,6	100,0
	Total	66	100,0	100,0	

coche3

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	17	25,8	25,8	25,8
	1	47	71,2	71,2	97,0
	2	2	3,0	3,0	100,0
	Total	66	100,0	100,0	

coche4

		Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válido	0	60	90,9	90,9	90,9
	1	6	9,1	9,1	100,0
	Total	66	100,0	100,0	