

Desarrollo del pensamiento computacional “desenchufado” mediante resolución de problemas: una revisión sistemática y meta-análisis

Development of “Unplugged” Computational Thinking through Problem Solving: Systematic Review and Meta-analysis

Natalia Moreno Palma¹, Francisco Javier Hinojo Lucena², José María Romero Rodríguez³, Magdalena Ramos Navas-Parejo⁴

¹ Universidad de Granada nmoreno@ugr.es

² Universidad de Granada fhinojo@ugr.es

³ Universidad de Granada romejo@ugr.es

⁴ Universidad de Granada magdalena@ugr.es

Recibido: 19/1/2024

Aceptado: 23/9/2024

Copyright ©

Facultad de CC. de la Educación y Deporte.
Universidad de Vigo



Dirección de contacto:

José María Romero Rodríguez
Facultad de Ciencias de la Educación
Campus Universitario de Cartuja
18071 Granada

Resumen

La creciente demanda de profesiones tecnológicas ha impulsado la educación STEAM y el desarrollo del pensamiento computacional desde edades tempranas. Aunque enseñar programación se considera eficaz para fomentar este tipo de pensamiento, se ha observado que centrarse exclusivamente en la codificación a corto plazo puede no ser la estrategia más adecuada. El objetivo de este trabajo fue analizar estudios empíricos que tratan sobre el desarrollo del pensamiento computacional en alumnado de diferentes etapas educativas. Para ello, se utilizó una metodología de revisión sistemática con meta-análisis, donde se analizaron 12 estudios realizados en etapas educativas desde Educación Infantil hasta Superior. Once de esos estudios reportaron evidencias significativas de aprendizaje del pensamiento computacional mientras que solamente uno de ellos no halló diferencias significativas. A su vez, el tamaño del efecto global fue significativo a favor del grupo experimental. Finalmente, este enfoque se adapta a las necesidades de los niños, ya que les permite aprender de manera lúdica y creativa, fomentando habilidades cognitivas, lógicas y de resolución de problemas.

Palabras clave

Pensamiento Computacional, Pensamiento Computacional Desenchufado, Resolución de Problemas, Recursos Educativos, Prácticas Educativas

Abstract

The increasing demand for technological professions has boosted STEAM education and the development of computational thinking from an early age. Although teaching programming is considered effective in fostering this type of thinking, it has been observed that focusing exclusively on coding in the short term may not be the most appropriate strategy. The aim of this work was to analyze empirical studies that deal

with the development of computational thinking in students of different educational stages. For this purpose, a systematic review with meta-analysis methodology was used, where 12 studies conducted in educational stages from preschool to higher education were analyzed. Eleven of these studies reported significant evidence of learning computational thinking while only one of them found no significant differences. In turn, the overall effect size was significant in favour of the experimental group. Finally, this approach adapts to the needs of children, as it allows them to learn in a playful and creative way, fostering cognitive, logical, and problem-solving skills.

Key Words

Computational Thinking, Computational Thinking Unplugged, Problem Solving, Educational Resources, Educational Practices

1. INTRODUCCIÓN

La potencialidad de los medios computacionales a nuestra disposición ofrece un panorama en el que se ha vuelto necesario capacitarse para profesiones de ámbito informático. Es por este motivo por lo que algunas de las profesiones más demandadas en la sociedad global y cambiante en la que nos encontramos requieren habilidades y capacidades que permitan trabajar en ambientes virtuales y digitales, el manejo de datos a gran escala, de software específicos o de la programación (Abesadze y Nozadze, 2020).

Como respuesta a esta demanda profesional, los sistemas educativos de diferentes países han comenzado a fomentar el desarrollo de competencias asociadas a disciplinas técnicas (Mouza et al., 2020; Romero-Rodríguez et al., 2021; Saad y Zainudin, 2022; Segredo et al., 2017). Estas competencias, habilidades y destrezas que caracterizan a profesionales de la computación, ingenieros o técnicos y que capacitan para el desempeño de profesiones estrechamente ligadas a la tecnología, describen una forma de pensar, un pensamiento específico: el pensamiento computacional (Ezeamuzie y Leung, 2022).

Teniendo en cuenta que el dominio de la programación y la codificación es la más clara de las señales que indican un control de habilidades, destrezas y capacidades que conforman el pensamiento computacional, plantearse enseñar estas disciplinas desde las primeras etapas educativas es una de las soluciones más comunes que podemos encontrar en la literatura, enfoque respaldado por la teoría conductista del aprendizaje (Belmar, 2022; Buitrago et al. 2017; Fagerlund et al. 2021; Palts y Pedaste, 2020).

A pesar de que enseñar a programar puede parecer ideal para mejorar el pensamiento computacional, múltiples estudios sugieren que aprender lenguajes de programación específicos no garantiza su desarrollo integral (Buitrago et al, 2017; Kandemir et al., 2021). Sin embargo, la programación puede ser efectiva para este propósito si se aborda desde un enfoque constructivista, enfocándose en una instrucción a largo plazo que tenga en cuenta las características del aprendizaje, más que centrada en el aprendizaje de lenguajes específicos a corto plazo (Papavlasopoulou et al., 2019).

Por esta razón, el desarrollo del pensamiento computacional mediante la enseñanza de la programación o codificación a estudiantes inexpertos siguiendo el paradigma constructorista requiere una cadena de logros cognitivos para los que se necesita tiempo y recursos personales muy concretos que dificultan este tipo de puesta en práctica (Buitrago et al., 2017; Fuentes-Rosado y Moo-Medina, 2017; Mouza et al., 2020).

Por otro lado, diversos estudios han explorado el uso de tecnologías avanzadas como la robótica, la realidad aumentada y la inteligencia artificial para fomentar el pensamiento computacional (Aristawati et al., 2018; Lin et al, 2021; Mejía et al., 2022; Witherspoon et al., 2017). A pesar de su potencial, estas herramientas pueden tener restricciones debido a su costo y la necesidad de formación especializada para los docentes (Cerón, 2022).

Alternativamente, el enfoque construccionista del aprendizaje del pensamiento computacional sugiere que se pueden establecer las bases de este tipo de pensamiento desde edades tempranas sin depender de computadoras (Montes-León et al., 2020; Threekunprapa y Yasri, 2020). Bajo este enfoque, se prioriza la observación, manipulación o la resolución de problemas y una vez llegado el momento, se introducen conceptos relacionados con la codificación o programación, previo desarrollo de las habilidades necesarias para su correcto desempeño (Miranda Pinto y Osório, 2019).

Este planteamiento de la enseñanza se presenta como opción factible de ser llevada a las aulas pues se han observado resultados sólidos de aprendizaje siguiendo este enfoque y, además, no se precisarían recursos educativos que supongan un coste elevado para los centros ni personal específicamente cualificado en computación en todas las etapas educativas (Bell y Vahrenhold, 2018; Hromkovic y Staub, 2019; Zapata-Ros, 2019).

A la luz de las aportaciones teóricas, se planteó el objetivo de analizar los estudios empíricos que tratan sobre el desarrollo del pensamiento computacional en alumnado de diferentes etapas educativas, determinando aquellas investigaciones que utilicen metodologías de resolución de problemas y materiales o recursos desenchufados, es decir, que no precisen de computadoras para su puesta en práctica.

Las preguntas de investigación que, junto con este objetivo principal, guiaron el presente estudio fueron las siguientes:

- RQ1: ¿Cuáles son las características más destacadas de los estudios (distribución geográfica y temporal, tipologías de publicación, procedimientos de selección de muestras y configuración de grupos, características de las muestras, instrumentos de evaluación y diseños metodológicos) que se han desarrollado sobre este tipo de intervención con estudiantes?
- Q2: ¿Cuáles son las características más relevantes de las intervenciones (ámbito del conocimiento en el que se desarrollan, procedimientos, prácticas, estrategias, técnicas y recursos de intervención) que se han implementado?
- Q3. ¿Cuál es la evidencia significativa de los estudios para generar mejoras en el desarrollo del pensamiento computacional en el estudiantado?

2. MARCO TEÓRICO

Se realizó una Revisión Sistemática de la Literatura (SLR) con meta-análisis, con el propósito de obtener un análisis detallado, selectivo, crítico y estructurado del tópico seleccionado (Page et al., 2021a; Page et al., 2021b). A su vez, se utilizó la estructura formal del marco PICOC para definir el objetivo y el ámbito de la revisión (Petticrew y Roberts, 2008). Dado que el objetivo de este estudio no fue comparar dos tipos de intervenciones, se ajustó la búsqueda al acrónimo PIOC y, en consecuencia, el objetivo se tradujo de la siguiente manera: analizar qué intervenciones basadas en una metodología de resolución de problemas con recursos o materiales “desenchufados” (I) se han

implementado para favorecer el pensamiento computacional (O) en estudiantes (P) de diferentes niveles educativos (C).

La búsqueda de producción se realizó basándose en dos conceptos: “pensamiento computacional” y “resolución de problemas”. Además, para centrar la búsqueda en el campo de la educación se incluyeron los términos “educación” y “estudiantes”. Puesto que el término “unplugged” o “desconectado” no se trata de un concepto normalizado o estandarizado, se decidió no incluirlo en la ecuación de búsqueda para evitar sesgos en el sondeo de la literatura. La variable independiente en esta investigación fue el proceso de instrucción desconectado del pensamiento computacional y la variable dependiente, el pensamiento computacional.

Para formular la ecuación de búsqueda, se llevó a cabo un análisis temático de los términos centrales de búsqueda utilizado para ello el Tesauro de la UNESCO (Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura [UNESCO], 2022) y el Tesauro Europeo de la Educación (Educational Resources Information Center [ERIC], 2022). Teniendo en cuenta el análisis PIOC previo, los términos principales y los seleccionados finalmente se muestran en la Tabla 1.

Población	Intervención	Resultados	Contexto
Students	Problem solving	Computational thinking	Education
Términos añadidos			
Post-secondary students			Basic education
University students			Adult education
Postgraduates			Early childhood education
Primary school students			Higher education
Secondary school students			Special needs education
Undergraduates			Formal education
Students Teachers			Secondary education
			Vocational education
			Professional education
			Technical education

Tabla 1. Términos de búsqueda centrales y términos añadidos tras llevar a cabo el análisis temático

Una vez identificados los términos que forman parte de la estrategia de búsqueda se planteó la ecuación de búsqueda que combina los conceptos seleccionados usando lógica booleana. La ecuación canónica que se adaptó a cada fuente de recogida de datos es la siguiente: “problem solving” AND “computational thinking” AND (students OR postgraduates OR undergraduates OR teachers OR education). Con respecto a las fuentes de datos en las que se ha llevado a cabo la investigación, se han seleccionado las bases de datos Web of Science (Web of Science Core Collection, KCI-Korean Journal Database, Current Contents Connect, ProQuest Dissertations & Theses Citation Index, MEDLINE, Preprint Citation Index, SciELO Citation Index, DIOSIS Citation Index, BIOSIS Previews y Derwent Innovations Index) y Scopus. La disciplina y la exigencia en la que se enmarca esta revisión justifica su idoneidad, así como el acceso online y el amplio alcance que ofrecen para su consulta.

Tras definir las preguntas y los objetivos de investigación, la ecuación de búsqueda y las fuentes de datos, se definieron criterios de inclusión y exclusión atendiendo a las premisas de la declaración PRISMA (Page et al., 2021a; Page et al., 2021b). Se tuvieron en cuenta los siguientes aspectos:

- Marco temporal: basándose en la relativa actualidad de la temática de la revisión que tiene su origen a comienzos del siglo XXI, se incluyeron investigaciones llevadas a cabo hasta 2022. No se incluyó 2023 por no ser un año finalizado en el momento de realizar la revisión.
- Tipología de estudios: los estudios seleccionados fueron artículos científicos revisados por pares para garantizar la calidad de la investigación
- Idioma: para la correcta selección e interpretación de los datos ofrecidos en los artículos seleccionados por parte de los investigadores, se incluyeron artículos escritos en inglés y español.
- Accesibilidad: con el objetivo de basar esta revisión en una metodología transparente y replicable, se seleccionaron artículos de fuente abierta.
- Metodología: se incluyeron estudios empíricos con diseño preexperimental, experimental o cuasiexperimental en los que se especificaran claramente los objetivos de investigación y las metodologías utilizadas; que respondiesen adecuadamente a las preguntas de investigación planteadas; presentaran datos sobre la evaluación de la intervención llevada a cabo y se describiese la misma; en los que se especificaran los recursos y materiales utilizados y fuese clara la relación entre éstos y la interpretación de los datos o las conclusiones obtenidas.
- Temática: se seleccionaron aquellos artículos en los que la intervención llevada al aula no hacía uso de un lenguaje de programación con compilador, de un entorno de desarrollo de software o de programas o aplicaciones softwares. A su vez se seleccionaron artículos en los que la variable dependiente de la investigación fuese el pensamiento computacional (evaluado de forma holística o a partir de algunas de sus componentes) y la variable independiente fuese el proceso educativo llevado a cabo en la intervención de aula.

En conclusión, los criterios de inclusión y exclusión se resumen en la Tabla 2.

Criterios de inclusión (IN)	Criterios de Exclusión (EX)
IN1: Artículos científicos	EX1: Documentos no revisados por pares
IN2: Publicaciones hasta diciembre de 2022	EX2: Artículos no escritos en inglés, francés o español.
IN3: Artículos escritos en inglés, francés o español	EX3: Acceso restringido
IN4: Acceso abierto	EX4: Investigaciones no empíricas
IN5: Investigaciones empíricas con diseño preexperimental, experimental o cuasiexperimental	EX5: Investigaciones que usen metodologías educativas, materiales y recursos didácticos enchufados o no tengan por temática desarrollar el pensamiento computacional.
IN6: Investigaciones que usen metodologías educativas, materiales desenchufados y recursos didácticos desenchufados para desarrollar el pensamiento computacional.	

Tabla 2. Criterios de inclusión y exclusión

Una vez finalizada la fase de planificación del proceso de revisión, se procedió con la fase de acción. Siguiendo el protocolo PRISMA (Page et al., 2021a; Page et al., 2021b), se efectuó el sondeo de la literatura usando la ecuación de búsqueda canónica adaptada a cada base de datos y se ejecutaron diferentes consultas hasta obtener los registros finales. El número de registros obtenidos en la última consulta puede verse en la Tabla 3.

Ecuación de búsqueda para Web of Science	Registros
TS= ('problem solving' AND 'computational thinking' AND (student* OR postgraduate* OR undergraduate* OR teacher* OR education))	768
Ecuación de búsqueda para Scopus	Registros
(TITLE-ABS-KEY ('problem solving') AND TITLE-ABS-KEY ('computational thinking') AND TITLE-ABS-KEY (student* OR postgraduate* OR undergraduate* OR teacher* OR education))	795

Tabla 3. Registros obtenidos en las bases de datos Web of Science y Scopus

Se utilizó Refworks como herramienta de gestión bibliográfica para organizar la información de los estudios obtenidos en las búsquedas y se empleó una plantilla de selección previamente confeccionada en una hoja de extracción de datos (Excel) para registrar los motivos de exclusión de cada estudio. Además, se utilizó otra plantilla para codificar y extraer datos de los estudios seleccionados para la revisión final. Todas las acciones implementadas durante el proceso de selección de los estudios (representado en la Figura 1) se llevaron a cabo manualmente sin usar herramientas automatizadas.

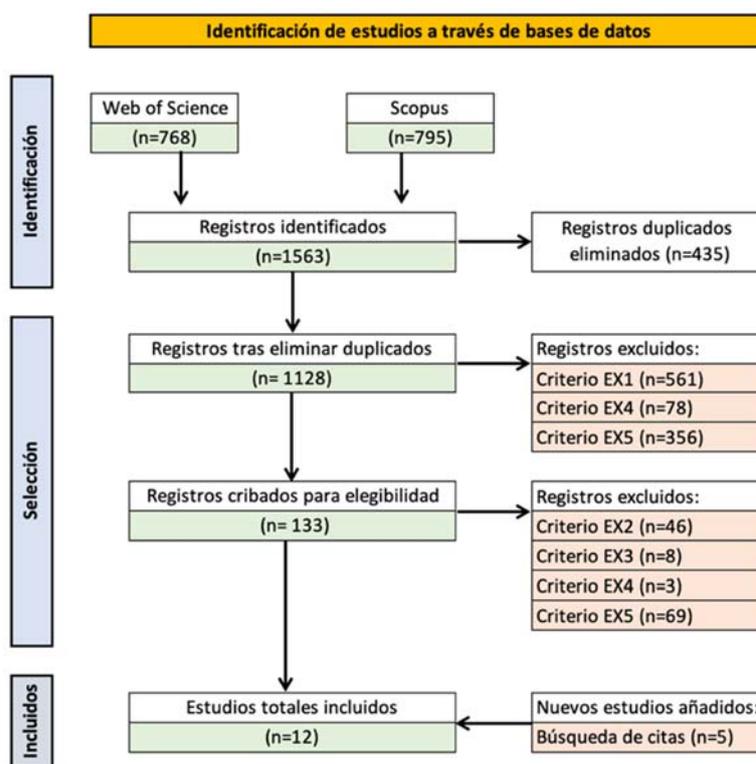


Figura 1. Diagrama de flujo

Una vez confirmada la muestra final de estudios, se extrajeron y codificaron los datos y la información de cada estudio en torno a las siguientes variables: (a) características contextuales y descripción de la muestra (país, procedimiento de selección de muestras, tamaño de la muestra, edad de la muestra, sexo de la muestra y nivel educativo de la muestra); (b) características metodológicas (diseño metodológico, instrumentos de evaluación usados para medir las variables dependientes y tipo de análisis de datos); (c) características de la variable independiente (el entorno en el que se desarrolla la intervención, áreas o componentes de la intervención, duración, procedimientos,

prácticas, estrategias, técnicas y/o recursos de la intervención); y (d) variables dependientes y resultados (componentes del pensamiento computacional analizados y resultados obtenidos).

Finalmente, para el meta-análisis se utilizó la medida de diferencia de medias estandarizada como indicador de los resultados (Viechtbauer, 2010). Se aplicó un modelo de efectos aleatorios a los datos, tal como se muestra en la Tabla 4. Para evaluar el grado de heterogeneidad (τ^2), además de calcular esta medida, se llevó a cabo la prueba Q para detectar heterogeneidad y se estimó el estadístico I^2 . Cuando se identificó cierto nivel de heterogeneidad ($\tau^2 > 0$, sin importar los resultados de la prueba Q), se estableció un intervalo de predicción para estimar con mayor precisión los resultados reales (Tabla 5). Para detectar estudios atípicos o influyentes en el modelo, se analizaron los residuos estandarizados y las distancias de Cook. Los estudios con residuos estandarizados superiores al percentil $100 \times (1 - .05/(2 \times k))$ de una distribución normal estándar se consideraron posibles valores atípicos. En este caso, se aplicó una corrección de Bonferroni con un nivel alfa de ,05 de dos colas, dado que el metaanálisis incluyó seis estudios. Para identificar estudios influyentes, se seleccionaron aquellos con distancias de Cook que excedían la mediana más seis veces el rango intercuartílico de las mismas. La asimetría del diagrama de embudo se evaluó mediante la prueba de correlación de rangos y la prueba de regresión, utilizando el error estándar de los resultados como predictor (Figura 2). Todo el análisis se llevó a cabo utilizando el software Jamovi, versión 2.3.

	Estimación	Error estándar	Z	p	CI Límite inferior	CI Límite superior
Intercepción	,625	,207	3,02	,003	,220	1,030

Tabla 3. Modelo de efectos aleatorios. Nota: estimador Tau²: máxima verosimilitud restringida

Tau	Tau ²	I ²	H ²	R ²	df	Q	p
,424	,1799 (SE = ,1618)	71,36%	3,491	.	5,000	16,231	,006

Tabla 4. Estadísticas de heterogeneidad

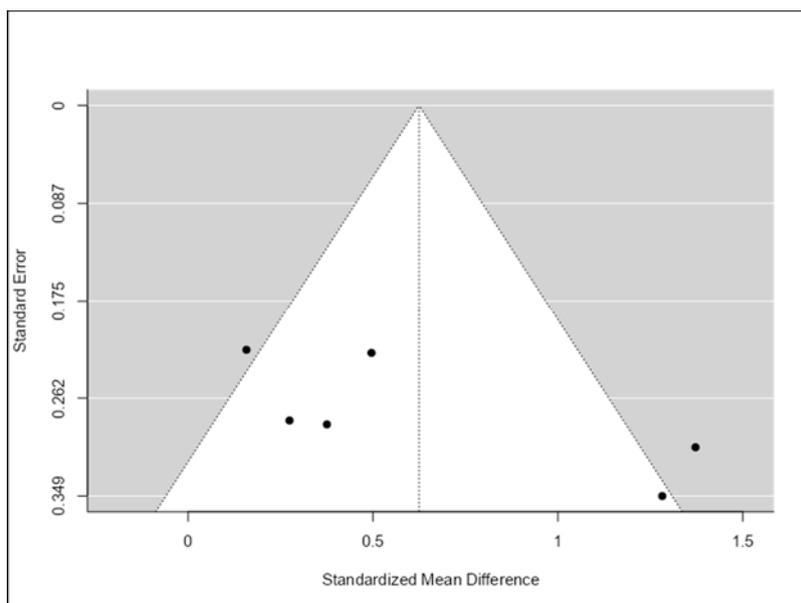


Figura 2. Diagrama de embudo

3. INFERENCIAS

Los estudios seleccionados procedieron de Chipre (N=1), Corea del Sur (N=2), España (N=3), Italia (N=1), Turquía (N=4) y Uruguay (N=1). En términos de la muestra total, en estos estudios se examinaron 764 participantes. El tamaño de la muestra osciló entre 11 y 131 participantes. La distribución de sexo no varió entre los estudios, las 12 investigaciones incluyeron muestras compuestas tanto por hombres como por mujeres. No se especificaron las edades de los participantes de todos los estudios, pero las etapas educativas en las que se centraron fueron Educación Infantil (N=5), Educación Primaria (N=6) y Educación Superior (N=1). Todas las investigaciones utilizaron un muestreo no probabilístico.

En relación a los instrumentos de evaluación, dos estudios utilizaron la escala estandarizada "Computational Thinking Scale" (CTS) (Polat y Yilmaz, 2022; Tonbuloglu y Tonbuloglu, 2019), un estudio empleó el test estandarizado "Test de Pensamiento Computacional" (TPC) (Merino-Amero et al., 2022) y el resto de los estudios utilizaron instrumentos ad hoc (Angeli y Valanides, 2020; Delal y Oner, 2020; Del Olmo-Muñoz et al., 2020; García-Valcárcel y Caballero-González, 2019; Gerosa et al., 2022; Nam et al., 2019; Peretti et al., 2020; Shim et al., 2016; Uzumcu y Bay, 2020). Finalmente, dos estudios utilizaron análisis de datos tanto cuantitativos como cualitativos (Tonbuloglu y Tonbuloglu, 2019; Uzumcu y Bay, 2020) y los diez estudios restantes utilizaron únicamente metodologías de análisis de datos cuantitativos (Angeli y Valanides, 2020; Delal y Oner, 2020; Del Olmo-Muñoz et al., 2020; García-Valcárcel y Caballero-González, 2019; Gerosa et al., 2022; Merino-Amero et al., 2022; Nam et al., 2019; Peretti et al., 2020; Polat y Yilmaz, 2022; Shim et al., 2016).

En cuanto al diseño de la investigación, tres estudios utilizaron un diseño preexperimental pretest-posttest (Delal y Oner, 2020; Tonbuloglu y Tonbuloglu, 2019; Shim et al., 2016), ocho estudios incluyeron un diseño cuasi experimental pretest-posttest, con un diseño de grupo control no equivalente (Angeli y Valanides, 2020; Del Olmo-Muñoz et al., 2020; García-Valcárcel y Caballero-González, 2019; Gerosa et al., 2022; Merino-Amero et al., 2022; Nam et al., 2019; Peretti et al., 2020; Polat y Yilmaz, 2022) y un estudio utilizó un diseño de programa inverso pretest-posttest con dos grupos experimentales (uno por cada intervención llevada a cabo) (Uzumcu y Bay, 2020).

En lo que respecta a las características de las intervenciones, en la Tabla 6 se muestran las principales estrategias y técnicas que se utilizaron durante su desarrollo, la duración de las intervenciones, las componentes del pensamiento computacional estudiadas (o bien el pensamiento computacional de una forma holística) y los instrumentos de recogida de información empleados. La mitad de los estudios se valieron de recursos robóticos (N=6), otros utilizaron la metodología de resolución de problemas basándose en tareas de The Kesfet Projet (www.kesfetprojesi.org) (N=1), de Code.org (<https://code.org/>) (N=1), Bebras (<https://www.bebas.org/>) (N=1) y del currículo escolar vigente para la etapa (N=3).

Referencia	Componentes	Duración	Procedimientos, materiales y recursos	Instrumentos
Angeli y Valanides (2020)	Pensamiento computacional (holística), relaciones espaciales, depuración, descomposición, intentos fallidos antes del éxito	2 sesiones de 40 min por participante	Resolución de problemas contextualizados a través de una historia y basados en robótica educativa (Bee-bot y tapetes cuadriculados). Uso de dos sistemas de memoria externos diferentes en los grupos control y experimental como apoyo del aprendizaje, (tarjetas o notas escritas). Trabajo individual o colaborativo con el investigador.	Ad hoc
Delal y Oner (2020)	Pensamiento computacional (holística)	2 semanas (5 sesiones de 1 h)	Resolución de problemas basados en desafíos Bebras. Trabajo individual o en grupos.	Ad hoc
Del Olmo-Muñoz et al. (2020)	Pensamiento computacional (holística)	8 semanas (8 sesiones de 45 min)	Resolución de problemas desconectados en el grupo experimental y conectados en el grupo control basados en programas de la plataforma Code.org.	Ad hoc
García-Valcárcel y Caballero-González (2019)	Secuencias, correspondencia acción-instrucción, depuración	8 sesiones (4 h por sesión)	Resolución de problemas basados en robótica educativa. Trabajo colaborativo en grupos. Uso del robot Bee-bot y tapetes cuadriculados. No se especifica la intervención llevada a cabo en el grupo control.	Ad hoc
Gerosa et al. (2022)	Pensamiento computacional (holística)	11 sesiones de 30 min	Resolución de problemas basados en robótica educativa. Trabajo en grupo. Uso del robot RoboTito, tapetes cuadriculados y tarjetas de colores. Uso de dispositivos electrónicos en el grupo control y objetos del entorno en el experimental para programar el robot.	Ad hoc
Merino-Amero et al. (2022)	Pensamiento computacional (holística)	6 semanas (18 sesiones de 45 min)	Resolución de problemas contextualizados en el área de Ciencias Sociales integrando el pensamiento computacional en las tareas del grupo experimental.	TPC
Nam et al. (2019)	Secuenciación, resolución de problemas	8 semanas	El grupo experimental siguió una intervención de 4 fases relacionadas con la resolución de problemas. Se hizo uso de robótica educativa (Turteblot y tapetes cuadriculados), tableros numéricos y puzzles. El grupo control utilizó resolución de problemas enmarcados en el currículo de la etapa.	Ad hoc
Peretti et al. (2020)	Habilidades visuales-espaciales, habilidades programación secuencial	4 semanas (3 sesiones de 1 h)	Resolución de problemas contextualizados a través de una historia basados en robótica educativa. Uso del robot Cubetto y tapetes cuadriculados. El grupo experimental y el grupo control fueron evaluados al mismo tiempo, pero el grupo control siguió la intervención tras el grupo experimental.	Ad hoc
Polat y Yilmaz, (2022)	Pensamiento computacional (holística)	7 semanas (14 sesiones de 1 h)	Resolución de problemas desconectados en el grupo experimental y conectados en el grupo control basados en el currículo vigente. Trabajo en grupo, en parejas o individual.	CTS, Ad hoc
Shim et al. (2016)	Repetición, condición, bifurcación, parámetros, secuencia, función	Una sesión (8 h)	Resolución de problemas basados en robótica educativa. Ludificación. Uso de robots programables mediante interfaz táctil y tableros de juego.	Ad hoc
Tonbuloglu y Tonbuloglu (2019)	Resolución de problemas, creatividad, pensamiento algorítmico, colaboración y pensamiento crítico	10 semanas (2 h semanales)	Resolución de problemas basados en The Kesfet Proje. Trabajo en grupo, por pares o en gran grupo. Uso de torre de Hanoi, tangram y puzzles.	CTS, ad hoc
Uzumcu y Bay (2020)	Comprensión del problema, operadores, bucles, condiciones, diagramas de flujo, descomposición, abstracción, patrón, algoritmo, depuración	2 años (40 h durante dos semanas el primer año y 52 h durante 4 semanas el segundo año)	Resolución de problema abierto. Aprendizaje por descubrimiento. Lluvia de ideas. Discusión socrática. Juego de roles. Trabajo en grupo, por pares o en gran grupo.	Ad hoc

Tabla 6. Características de cada variable independiente

Por otro lado, teniendo en cuenta los resultados analizados de desarrollo del pensamiento computacional y de las variables vinculadas, siete estudios evaluaron el pensamiento computacional de una forma holística (Angeli y Valanides, 2020; Delal y

Oner, 2020; Del Olmo-Muñoz et al., 2020; Gerosa et al., 2022; Merino-Armero, 2022; Tonbuloglu y Tonbuloglu, 2019), y uno de ellos no obtuvo diferencias significativas entre grupo control y experimental (Polat y Yilmaz, 2022). Para la variable “secuenciación” (o descomposición), se encontraron un total de cinco estudios (García-Valcárcel y Caballero-González, 2019; Nam et al., 2019; Peretti et al., 2020; Uzumcu y Bay, 2020; Shim et al., 2016), uno de ellos sin resultados de aprendizaje estadísticamente significativos al comparar los resultados del pretest y postest (Shim et al., 2016). Además, se encontraron dos estudios que mostraron diferencias significativas entre grupos o resultados de aprendizaje estadísticamente significativos para las variables “depuración” (García-Valcárcel y Caballero-González, 2019; Uzumcu y Bay, 2020), “condición” y “bucles” (Shim et al., 2016; Uzumcu y Bay, 2020) y “algoritmia” (Tonbuloglu y Tonbuloglu, 2019; Uzumcu y Bay, 2020). A su vez, se encontró un único estudio que reportó evidencias significativas de aprendizaje para las variables “creatividad”, “colaboración”, “pensamiento crítico” (Tonbuloglu y Tonbuloglu, 2019), “abstracción y patrón” (Uzumcu y Bay, 2020) y “parámetros” (Shim et al., 2016). También se encontró un único artículo para la variable “función” sin resultados estadísticamente significativos (Shim et al., 2016). Por último, tres estudios consideraron la variable “resolución de problemas” (Nam et al., 2019; Uzumcu y Bay, 2020; Tonbuloglu y Tonbuloglu, 2019) de los cuales uno de ellos no reportó diferencias significativas entre pretest y postest (Tonbuloglu y Tonbuloglu, 2019).

Finalmente, las diferencias de medias estandarizadas encontradas presentaron un rango que iba desde ,1580 a 1,3724, con una clara mayoría de estimaciones positivas (100% de las observaciones). El modelo de efectos aleatorios mostró una diferencia media estandarizada estimada de ,6247 (IC del 95%: ,2196 a 1,0298), por lo que el resultado medio difiere significativamente de cero ($z = 3,0225$, $p = ,0025$). La prueba Q evidenció una heterogeneidad considerable en los resultados ($Q(5) = 16,2305$, $p = ,0062$, $\tau^2 = ,1799$, $I^2 = 71,3586\%$). Al observar el intervalo de predicción de los resultados reales (IC del 95%: -,3001 a 1,5495), se destacó que, aunque la media estimada fuera positiva, algunos estudios podrían arrojar resultados negativos. El análisis de los residuos estandarizados mostró que ninguno de los estudios tenía un valor superior a $\pm 2,6383$, por lo que no existen indicios de valores atípicos en el contexto de este modelo. Según las distancias de Cook, ninguno de los estudios se considera excesivamente influyente. La prueba de regresión indicó asimetría en el diagrama de embudo ($p = ,0288$), pero no la prueba de correlación de rangos ($p = ,1361$). El tamaño del efecto global fue pequeño, con un d de Cohen de ,22, lo que sugiere que la diferencia entre los grupos es mínima.

El diagrama de bosque incluyó los seis estudios que recopilaron información sobre medias y desviaciones estándar para los grupos de control y experimental en relación al pensamiento computacional (Figura 3).

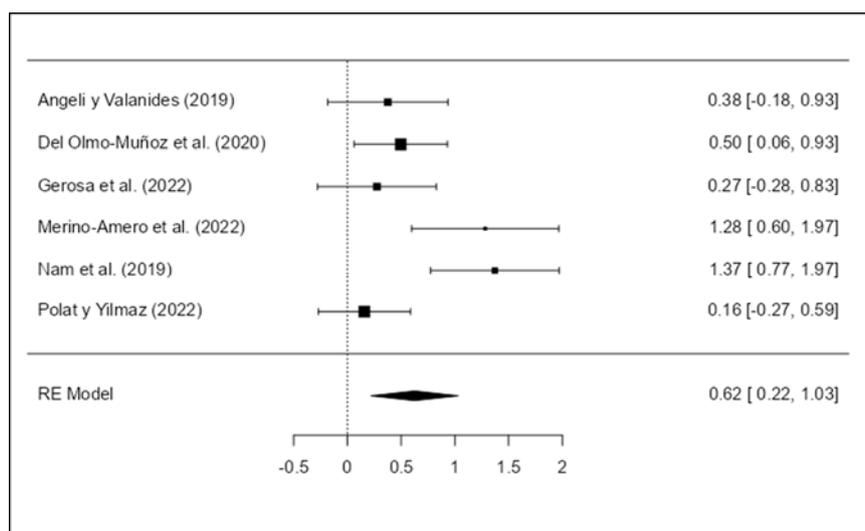


Figura 3. Diagrama de bosque

4. CONCLUSIONES

El propósito de esta revisión sistemática con meta-análisis fue sintetizar las principales características de las prácticas docentes que tienen por objetivo el desarrollo del pensamiento computacional a través de la aplicación de una metodología basada en la resolución de problemas, utilizando recursos o materiales que no requieren de conexión a dispositivos electrónicos.

En relación a la primera pregunta de investigación, los resultados obtenidos revelan que los estudios empíricos relacionados con esta temática han sido realizados en diversos países, incluyendo Turquía, Uruguay, Corea del Sur, Chipre, Italia y España. Estos estudios abarcan un periodo de tiempo relativamente reciente, comprendido entre los años 2016 (N=1), 2019 (N= 3), 2020 (N=5) y 2022 (N=3). La concentración de estudios durante este período sugiere una mayor conciencia en la importancia del pensamiento computacional en la educación y la búsqueda de estrategias efectivas para su enseñanza, lo que podría tener implicaciones significativas en la forma en que se aborda la educación en el siglo XXI (Segredo et al., 2017; Saad y Zainudín, 2022; Palts y Pedaste, 2020).

La cantidad de artículos incluidos en esta revisión puede considerarse limitada, dadas las diversas áreas en las que el pensamiento computacional puede tener aplicaciones, su estrecha vinculación con la tecnología y las matemáticas, así como su actual relevancia en los planes de estudio educativos vigentes. Esta limitación se debe principalmente a la exclusión de numerosos estudios durante el proceso de selección de este trabajo por el uso de recursos, programación o codificación enchufada. De acuerdo con Zapata-Ros (2015), Buitrago et al. (2017), Belmar (2022), Fagerlund et al. (2021), Cerón (2022) y Kandemir et al., (2021), el principal enfoque actual de las investigaciones llevadas a cabo que tienen por finalidad desarrollar el pensamiento computacional, utilizan recursos enchufados.

Los resultados obtenidos en esta revisión sistemática resaltan el amplio uso de herramientas pedagógicas específicas en el desarrollo del pensamiento computacional desenchufado en diferentes niveles educativos. En el caso de la Educación Infantil, se observa una fuerte tendencia hacia la implementación de kits de robótica, que ofrecen a

los estudiantes una experiencia tangible y práctica en la comprensión de conceptos computacionales. Este recurso se utiliza principalmente para desarrollar el pensamiento computacional a través de un aprendizaje de los fundamentos de la codificación o programación en edades tempranas. Según señalan Cerón (2022), Buitrago et al. (2017), Aristawati et al. (2018) y Mejía et al. (2022) la adquisición de estas habilidades favorece de manera significativa el proceso de aprendizaje del pensamiento computacional. Esta afirmación se ve corroborada por los resultados de este análisis ya que siete de los 12 artículos seleccionados reportan resultados significativos de mejora del pensamiento computacional siguiendo este enfoque.

Por otro lado, en la Educación Primaria, se destaca el uso de recursos como puzzles, tarjetas, bloques y problemas impresos, que brindan oportunidades para la resolución de problemas sin necesidad de dispositivos electrónicos que supongan un coste elevado para los centros ni personal específicamente cualificado en computación (Mouza et al., 2020; Threekunprapa y Yasri, 2020; Montes-León et al., 2020; Hromkovic y Staub, 2019; Zapata-Ros, 2019; Bell y Vahrenhold, 2018; Cerón, 2022).

Es importante señalar que, en la etapa de Educación Primaria, los problemas contextualizados relacionados con la programación o la codificación se utilizan con frecuencia como herramientas efectivas para el desarrollo del pensamiento computacional. Sin embargo, a medida que se avanza en la Educación Superior, se observa una transición hacia problemas más abiertos y desafiantes.

A su vez, la mayoría de las intervenciones (N=11) se han implementado en las etapas de Educación Infantil y Primaria. Esta observación sugiere un enfoque constructivista en el proceso de enseñanza del pensamiento computacional desenchufado, que busca introducir conceptos computacionales desde edades tempranas en los estudiantes a través de la exploración y la resolución de problemas (Miranda Pinto y Osório, 2019; Papavlasopoulou et al., 2019; Belmar, 2022; Fagerlund et al. 2021).

No se han seleccionado estudios que versan sobre esta temática llevados a cabo en la etapa de Educación Secundaria, debido a que utilizan lenguajes de programación con compilador, entornos de desarrollo de software o aplicaciones software para desarrollar el pensamiento computacional en alumnado de Secundaria. La investigación sobre el desarrollo del pensamiento computacional en educación se inició centrada en la programación como herramienta principal y, posteriormente, se reconoció la existencia de diversas formas de promover el desarrollo del pensamiento computacional sin recurrir exclusivamente a la programación o codificación (Bell y Vahrenhold, 2018; Hromkovic y Staub, 2019; Montes-León et al., 2020; Papavlasopoulou et al., 2019; Threekunprapa y Yasri, 2020; Zapata-Ros, 2015; Zapata-Ros, 2019). Esta podría ser otra de las razones por la que la mayor parte de las investigaciones seleccionadas en la presente revisión se han llevado a cabo en etapas educativas inferiores a la Educación Secundaria.

En cuanto a la tercera pregunta de investigación, respecto a la identificación de evidencias sobre la efectividad de las intervenciones desenchufadas para desarrollar el pensamiento computacional, de acuerdo con los resultados de las investigaciones de Hromkovic y Staub (2019), Zapata-Ros (2019), Bell y Vahrenhold (2018), Threekunprapa y Yasri (2020), Montes-León et al. (2020) y Papavlasopoulou et al. (2019), se puede afirmar que el uso de materiales desenchufados, recursos desenchufados y metodologías de resolución de problemas genera mejoras de aprendizaje en habilidades del pensamiento computacional. No obstante, la variabilidad de subdimensiones del pensamiento computacional analizadas en los diferentes estudios no es muy amplia

tendiéndose a seleccionar variables generales debido a las cortas edades de los participantes de las investigaciones.

Además, el número de participantes en los estudios oscila ente 11 y 131, el bajo número de participantes en los estudios puede ser un factor limitante a la hora de extraer y generalizar resultados pues éstos pueden verse alterados por la motivación individual de cada sujeto o por las características concretas del aprendizaje personal.

Por su parte, el meta-análisis evidenció que el tamaño del efecto global de las investigaciones que utilizaron materiales desenchufados mejoraron significativamente el pensamiento computacional de los estudiantes asignados al grupo experimental, tal y como se recoge en el diagrama de bosque.

Esta revisión sistemática arroja luz sobre el creciente interés y desarrollo del pensamiento computacional desenchufado, que se está convirtiendo en un campo de estudio en constante crecimiento. Un hallazgo significativo que emerge de este análisis es la implementación de estrategias de enseñanza de pensamiento computacional desenchufado en las primeras etapas del desarrollo individual, es decir, en la Educación Infantil y Primaria. Este enfoque pedagógico coincide de manera notable con el paradigma construccionista del aprendizaje del pensamiento computacional. Además, este enfoque se adapta especialmente bien a las necesidades de los niños, ya que les permite aprender de manera lúdica y creativa, fomentando habilidades cognitivas, lógicas y de resolución de problemas desde una corta edad.

No obstante, este trabajo presenta algunas limitaciones. La búsqueda bibliográfica, tanto la primaria como la complementaria, se completó en diciembre de 2022, por lo que no se incluyó literatura publicada después de esa fecha. Además, existe la posibilidad de que el sesgo del investigador haya influido en la selección final de los estudios, a pesar de los esfuerzos realizados para mitigar este efecto mediante el uso de rúbricas objetivas y claras en el proceso de selección.

Como posibles direcciones futuras de investigación, se sugiere la realización de un metaanálisis de los estudios recopilados para evaluar el tamaño del efecto general de la investigación. Además, sería valioso llevar a cabo revisiones sistemáticas específicas para investigar en profundidad los estudios que se centran en el desarrollo de los diversos componentes del pensamiento computacional de manera desenchufada.

BIBLIOGRAFÍA

- Abesadze, S. y Nozadze, D. (2020). Make 21st century education: The importance of teaching programming in schools. *International Journal of Learning and Teaching*, 6(3), 6. <http://dx.doi.org/10.18178/ijlt.6.3.158-163>
- Angeli, C. y Valanides, N. (2020). Developing young children's computational thinking with educational robotics: An interaction effect between gender and scaffolding strategy. *Computers in human behavior*, 105, 105954. <https://doi.org/10.1016/j.chb.2019.03.018>
- Aristawati, F.A., Budiyanto, C. y Yuana, R.A. (2018). Adopting educational robotics to enhance undergraduate students' self-efficacy levels of computational thinking. *Journal of Turkish Science Education*, 15(Special), 42-50. <http://dx.doi.org/10.12973/tused.10255a>
- Belmar, H. (2022). Review on the teaching of programming and computational thinking in the world. *Frontiers in Computer Science*, 4. <https://doi.org/10.3389/fcomp.2022.997222>

- Bell, T. y Vahrenhold, J. (2018). CS unplugged—how is it used, and does it work? En H.J. Böckenhauer, D. Komm y W. Unger (Eds.). *Adventures between lower bounds and higher altitudes: essays dedicated to Juraj Hromkovič on the occasion of his 60th birthday*, (pp. 497-521). Springer. https://doi.org/10.1007/978-3-319-98355-4_29
- Buitrago, F., Casallas, R., Hernández, M., Reyes, A., Restrepo, S. y Danies, G. (2017). Changing a generation's way of thinking: teaching computational thinking through programming. *Review of Educational Research*, 87(4), 834-860. <https://doi.org/10.3102/0034654317710096>
- Cerón, J.A. (2022). Lenguaje de programación para niños y niñas: perspectivas conectadas y desconectadas en la educación básica. *Revista Internacional de Pedagogía e Innovación Educativa*, 3(1), 45-66. <https://doi.org/10.51660/ripie.v3i1.108>
- Delal, H. y Oner, D. (2020). Developing middle school students' computational thinking skills using unplugged computing activities. *Informatics in Education*, 19(1), 1-13. <https://doi.org/10.15388/infedu.2020.01>
- Del Olmo-Muñoz, J., Cózar-Gutiérrez, R. y González-Calero, J.A. (2020). Computational thinking through unplugged activities in early years of Primary Education. *Computers & Education*, 150, 103832. <https://doi.org/10.1016/j.compedu.2020.103832>
- Educational Resources Information Center (20 de diciembre de 2022). *Tesaurus Europeo de la Educación*. <https://vocabularyserver.com/tee/es/index.php>
- Ezeamuzie, N.O. y Leung, J.S. (2022). Computational thinking through an empirical lens: A systematic review of literature. *Journal of Educational Computing Research*, 60(2), 481-511. <https://doi.org/10.1177/0735633121103315>
- Fagerlund, J., Häkkinen, P., Vesisenaho, M. y Viiri, J. (2021). Computational thinking in programming with Scratch in primary schools: A systematic review. *Computer Applications in Engineering Education*, 29(1), 12-28. <https://doi.org/10.1002/cae.22255>
- Fuentes-Rosado, J. I. y Moo-Medina, M. (2017). Dificultades de aprender a programar. *Revista Educación en Ingeniería*, 12(24), 76-82. <https://doi.org/10.26507/rei.v12n24.728>
- García-Valcárcel, A. y Caballero-González, Y.A. (2019). Robótica para desarrollar el pensamiento computacional en Educación Infantil. *Comunicar: Revista científica iberoamericana de comunicación y educación*, 32(59), 63-72. <https://doi.org/10.3916/C59-2019-06>
- Gerosa, A., Koleszar, V., Tejera, G., Gómez-Sena, L. y Carboni, A. (2022). Educational robotics intervention to foster computational thinking in preschoolers: Effects of children's task engagement. *Frontiers in Psychology*, 13, 904761. <https://doi.org/10.3389/fpsyg.2022.904761>
- Hromkovič, J. y Staub, J. (2019). Constructing Computational Thinking using CS Unplugged. *Constructivist Foundations*, 14(3), 353-355. Recuperado de: <https://bit.ly/3FwjUgm>
- Kandemir, C.M., Kalelioğlu, F. y Gülbahar, Y. (2021). Pedagogy of teaching introductory text-based programming in terms of computational thinking concepts and practices. *Computer Applications in Engineering Education*, 29(1), 29-45. <https://doi.org/10.1002/cae.22374>
- Lin, Y.S., Chen, S.Y., Tsai, C.W. y Lai, Y.H. (2021). Exploring computational thinking skills training through augmented reality and AIoT learning. *Frontiers in psychology*, 12, 640115. <https://doi.org/10.3389/fpsyg.2021.640115>
- Mejía, I., Ariel Hurtado, J., Zúñiga, R.F. y Salazar, B.G. (2022). Robótica educativa como herramienta para el desarrollo del pensamiento computacional. Una revisión de la literatura. *Revista Educación en Ingeniería*, 17(33), 68-78. <http://dx.doi.org/10.26507/rei.v17n33.1216>

- Merino-Armero, J.M., González-Calero, J.A., Cózar-Gutiérrez, R. y del Olmo-Muñoz, J. (2022). Unplugged activities in cross-curricular teaching: effect on sixth graders' computational thinking and learning outcomes. *Multimodal Technologies and Interaction*, 6(2), 13. <https://doi.org/10.3390/mti6020013>
- Miranda Pinto, M.S. y Osório, A.J. (2019). Aprender a programar en Educación Infantil: Análisis con la escala de participación. *Pixel-Bit. Revista de Medios y Educación*, 55, 133-156. <http://dx.doi.org/10.12795/pixelbit.2019.i55.08>
- Montes-León, H., Hijón-Neira, R., Pérez-Marín, D. y Montes-León, S.R. (2020). Mejora del Pensamiento Computacional en Estudiantes de Secundaria con Tareas Unplugged. *Education in the Knowledge Society (EKS)*, 21(24). <https://doi.org/10.14201/eks.23002>
- Mouza, C., Pan, Y.C., Yang, H. y Pollock, L. (2020). A multiyear investigation of student computational thinking concepts, practices, and perspectives in an after-school computing program. *Journal of Educational Computing Research*, 58(5), 1.029-1.056. <https://doi.org/10.1177/07356331209056>
- Nam, K.W., Kim, H.J. y Lee, S. (2019). Connecting plans to action: The effects of a card-coded robotics curriculum and activities on Korean kindergartners. *The Asia-Pacific Education Researcher*, 28(5), 387-397. <https://doi.org/10.1007/s40299-019-00438-4>
- Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (20 de diciembre de 2022). *Tesaurus de la Unesco*. <https://vocabularies.unesco.org/browser/thesaurus/es/>
- Page, M.J., McKenzie, J.E., Bossuyt, P.M., Boutron, I., Hoffmann, T.C., Mulrow, C.D., Dhamseer, L., Tetzlaff, J.M., Akl, E.A., Brennan, S.E., Chou, R., Glanville, J., Grimshaw, J.M., Hróbjartsson, A., Lalu, M.M., Li, T., Loder, E.W., Mayo-Wilson, E., McDonald, S., ... Moher, D. (2021a). The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *International journal of surgery*, 88, 105906. <https://doi.org/10.1016/j.ijssu.2021.105906>
- Page, M.J., Moher, D., Bossuyt, P.M., Boutron, I., Hoffmann, T.C., Mulrow, C.D., Shamseer, L., Tetzlaff, J.M., Akl, E.A., Brennan, S.E., Chou, R., Glanville, J., Grimshaw, J.M., Hróbjartsson, A., Lalu, M.M., Li, T., Loder, E.W., Mayo-Wilson, E., McDonald, S., ... McKenzie, J.E. (2021b). PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. *BMJ*, 372, 160. <https://doi.org/10.1136/bmj.n160>
- Palts, T. y Pedaste, M. (2020). A model for developing computational thinking skills. *Informatics in Education*, 19(1), 113-128. <http://dx.doi.org/10.15388/infedu.2020.06>
- Papavlasopoulou, S., Giannakos, M.N. y Jaccheri, L. (2019). Exploring children's learning experience in constructionism-based coding activities through design-based research. *Computers in Human Behavior*, 99, 415-427. <https://doi.org/10.1016/j.chb.2019.01.008>
- Peretti, G., Villani, D., Marangi, M., Pelizzari, F., Di Bruno, S., Guida, I., Marchetti, A., Riva, G., Rivoltella, P.C. y Massaro, D. (2020). Coding with me: exploring the effect of coding intervention on preschoolers' cognitive skills. *Annual Review of Cybertherapy and Telemedicine*, 18, 153-156. Recuperado de: <https://bit.ly/3s0xJ3z>
- Petticrew, M. y Roberts, H. (2008). *Systematic reviews in the social sciences: A practical guide*. John Wiley & Sons. <https://doi.org/10.1002/9780470754887>
- Polat, E. y Yilmaz, R.M. (2022). Unplugged versus plugged-in: examining basic programming achievement and computational thinking of 6th-grade students. *Education and Information Technologies*, 27(7), 9.145-9.179. <https://doi.org/10.1007/s10639-022-10992-y>
- Romero-Rodríguez, J.M., Aznar, I., Hinojo-Lucena, F.J. y Gómez-García, G. (2021). Uso de los dispositivos móviles en educación superior: relación con el rendimiento académico y la autorregulación del aprendizaje. *Revista Complutense de Educación*, 32(3), 327-335. <https://doi.org/10.5209/rced.70180>

- Saad, A. y Zainudin, S. (2022). A review of Project-Based Learning (PBL) and Computational Thinking (CT) in teaching and learning. *Learning and Motivation*, 78, 101802. <https://doi.org/10.1016/j.lmot.2022.101802>
- Segredo, E.M., Miranda, G. y León, C. (2017). Towards the education of the future: computational thinking as a generative learning mechanism. *Education in the knowledge society: EKS*, 18(2), 33-58. <http://dx.doi.org/10.14201/eks20171823358>
- Shim, J., Kwon, D. y Lee, W. (2016). The effects of a robot game environment on computer programming education for elementary school students. *IEEE Transactions on Education*, 60(2), 164-172. <http://dx.doi.org/10.1109/TE.2016.2622227>
- Threekunprapa, A. y Yasri, P. (2020). Unplugged Coding Using Flowblocks for Promoting Computational Thinking and Programming among Secondary School Students. *International Journal of Instruction*, 13(3), 207-222. <https://doi.org/10.29333/iji.2020.13314a>
- Tonbuloglu, B. y Tonbuloglu, İ. (2019). The effect of unplugged coding activities on computational thinking skills of middle school students. *Informatics in Education*, 18(2), 403-426. <https://doi.org/10.15388/infedu.2019.19>
- Uzumcu, O. y Bay, E. (2020). The effect of computational thinking skill program design developed according to interest driven creator theory on prospective teachers. *Education and Information Technologies*, 26(1), 565-583. <https://doi.org/10.1007/s10639-020-10268-3>
- Viechtbauer, W. (2010). Conducting meta-analyses in R with the metafor package. *Journal of Statistical Software*, 36, 1-48. <http://dx.doi.org/10.18637/jss.v036.i03>
- Witherspoon, E.B., Higashi, R.M., Schunn, C.D., Baehr, E.C. y Shoop, R. (2017). Developing computational thinking through a virtual robotics programming curriculum. *ACM Transactions on Computing Education*, 18(1), 1-20. <https://doi.org/10.1145/3104982>
- Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. *Revista de Educación a Distancia (RED)*, 46. Recuperado de: <https://bit.ly/45QJoQj>
- Zapata-Ros, M. (2019). Pensamiento computacional desenchufado. *Education in the Knowledge Society (EKS)*, 20, 18-29. https://doi.org/10.14201/eks2019_20_a18