

**Universidad de Extremadura**

**Facultad de Formación del Profesorado**

**MÁSTER EDUCACIÓN DIGITAL**

Facultad  
de Formación  
del Profesorado



**El pensamiento computacional en la educación obligatoria.**

**Una revisión sistemática de la literatura.**

**Trabajo Fin de Máster**

Presentado por Jesús Acevedo Borrega

Dirigido por Jesús Valverde Berrocoso

Ciencias de la Educación

Cáceres, 2016

DEDICADO A MI “ TITA ”,

PUES TU ORGULLO ES FUERZA PARA EL DÍA A DÍA

元

# Índice de contenido

1. Resumen.....	4
2. Abstract.....	5
3. Introducción.....	6
4. Justificación.....	8
5. Objetivos.....	11
5.1. Objetivo General.....	11
5.2. Objetivo Específico.....	11
6. Marco Teórico.....	12
6.1. Definición de pensamiento computacional.....	12
6.2. Evolución de su aplicación en la educación.....	16
6.3. El pensamiento computacional en los últimos diez años.....	18
7. Metodología.....	23
7.1. Formulación del problema.....	23
7.2. Búsqueda y localización de las fuentes de estudios primarios.....	24
7.3. Codificación de las fuentes.....	26
7.4. Análisis de los resultados.....	27
7.5. Síntesis general o conclusiones.....	27
8. Resultados.....	28
8.1. Análisis de la finalidad de los artículos.....	33
8.2. Análisis del rango académico al que van dirigidos los artículos.....	35
8.3. Metodologías o tipos de artículos encontrados.....	37
8.4. Propuestas de introducción del pensamiento computacional en el currículum.....	38
8.5. Experiencias reales en educación primaria.....	40
9. Conclusiones.....	43
10. Bibliografía.....	48
10.1. Referencias documentales objeto de la revisión.....	51

## **1. Resumen**

Las repercusiones del pensamiento computacional en todos los niveles de la Educación son innumerables. Es por ello que en los últimos diez años se aprecia un aumento de la producción científica que estudia el concepto en todos sus ámbitos. Una situación que ha promovido la realización de revisiones sistemáticas.

Las realizadas durante los últimos años comprenden hasta el año 2014. Por esta razón, realizar una revisión sistemática de todos los artículos científicos producidos en los tres últimos años permitirá continuar el trabajo realizado y aportar nuevos datos. Una nueva base que permitirá a la comunidad educativa comenzar o mejorar sus investigaciones sobre el desarrollo del pensamiento computacional en la educación formal.

Se seguirá como metodología un protocolo de investigación que nos permita realizar un análisis riguroso de la situación actual. Tras lo cual, procederemos a interpretar y evaluar toda la información accesible sobre dicha cuestión, siguiendo una serie de criterios establecidos para mantener la objetividad. En los resultados generales se obtuvieron 185 artículos científicos, de los cuales son seleccionados 45.

Este resultado general es expuesto a un análisis profundo con el objetivo de dar respuesta a cinco temas. A saber: finalidad de los artículos científicos, rango académico al que van dirigidos los artículos, metodologías usadas en los artículos seleccionados, propuestas de introducción del pensamiento computacional en el currículum y experiencias reales llevadas a cabo en el aula.

En los resultados observaremos cómo el número de publicaciones sigue en aumento y cómo las tendencias son muy similares a las obtenidas en revisiones anteriores. Además, conoceremos las últimas investigaciones sobre el currículum y la práctica real. Una investigación que supondrá una mejora de la calidad en los procesos iniciales de investigaciones futuras.

**Palabras clave:** Pensamiento computacional, educación, resolución de problemas, aprendizaje.

## **2. Abstract**

The effects of computational thinking in all the levels or stages of Education are countless. That is why, in the last ten years, an increase in scientific production exploring the concept in all its range can be appreciated. A situation that has promoted the realization of systematic reviews.

Those made over the past few years comprise up to 2014. For this reason, a systematic review of all scientific articles produced in the last three years will allow to continue their work and provide new information. A new foundation that will enable the educational community to start or improve their research on the development of computational thinking in formal education.

It will be applied a methodology based on a research protocol that allows us to make a rigorous analysis of the current situation. Afterwards, we will interpret and evaluate all the information available on this issue, following a series of criteria to maintain objectivity. In the overall results, 185 scientific articles were found, of which 45 were selected.

This overall result is exposed to a profound analysis in order to respond to five themes. Specifically: purpose of the scientific articles, the target population to which the articles are directed, methodologies used in the selected articles, proposals for the introduction of computational thinking in the curriculum and actual experiments carried out in the classroom.

The results will allow us to observe an increase in the number of articles published and confirm that trends are very similar to those obtained in previous reviews. Beside, we will know the last researches about the curriculum and real experiences. An investigation that will involve a quality improvement in the initial processes of future researches.

**Key Words:** Computational thinking, education, problem solving, learning.

### **3. Introducción**

En los últimos años se ha impulsado un movimiento educativo a nivel internacional relacionado con la introducción del «pensamiento computacional», la programación informática y la robótica en aulas de diferentes niveles. De hecho, algunos sistemas educativos están incluyendo nuevas competencias en sus currículos vinculados con estos temas ya que los responsables de las políticas educativas asumen la existencia de razones económicas, laborales, educativas, sociales y culturales para introducir tales contenidos.

Nacidas todas ellas de una realidad futura donde las palabras Volatilidad, Incertidumbre, Complejidad y Ambigüedad serán sus características principales, de sus siglas en inglés VUCA (Volatility, Unvertainty, Complexity y Ambiguity). Es un entorno acuñado por el ejército americano para describir situaciones y condiciones generales. Como muchos términos militares fue adaptado inicialmente al mundo empresarial y hoy en día es usado en el ámbito educativo, debido a las condiciones adaptativas extremas que implica el mismo. Un entorno que tiene como mayor implicación que el mundo actual y el venidero evolucionará cada vez a mayor velocidad, lo cual generará mayor dificultad, complejidad e imprevisibilidad.

[...] All such adaptation is essentially a process of learning; the organism that can master tomorrow what had yesterday defeated it may be said to have learned. [...] If conditions are changing more rapidly than the organism can learn (or adapt), it will fail; it may even die, or cease to exist. (*Revans, 1979, p. 2*)<sup>1</sup>

Si a esta realidad futura aplicásemos la Ley de Revans, entendiendo como organismos vivos a las personas ante un entorno tan veloz, implica que debemos aprender a la misma velocidad si nuestro deseo no es el de fracasar. Ambos, pues, nos imponen un aprendizaje continuo no anclado a la edad del discente.

---

<sup>1</sup> [...] Toda adaptación es esencialmente un proceso de aprendizaje, el organismo que puede dominar mañana lo que fue derrotado ayer, se puede decir que ha aprendido. [...] Si las condiciones cambian más rápidamente de lo que el organismo puede aprender (o adaptarse), fracasará, e incluso puede morir o dejar de existir. (*Revans, 1979, p. 2*).

Por ende, debemos focalizar nuestra atención en ¿qué debemos aprender durante nuestra vida en un entorno tan volátil? Las respuestas deberían ser aprendizajes que estuvieran al margen del paso del tiempo, un aprendizaje que ajeno a las herramientas o las metodologías, permita a los eternos discentes hacer frente a su futuro.

Mas la realidad descrita es muy cercana, dado que se estima que más de un millón de puestos de trabajo, sólo en los Estados Unidos, quedarán vacantes en los próximos años porque los ciudadanos del futuro carecerán de las competencias necesarias para el uso eficaz de las Tecnologías Digitales. Por tanto, la educación básica debe asumir la necesidad de redefinir qué son los imprescindibles para la futura sociedad. Entre los cuales, sin duda estará la capacidad de conocer, comprender y aplicar el pensamiento computacional, un elemento atemporal.

Eso sí, nada implica que ese aprendizaje se realice de un modo aislado sino, todo lo contrario, en un contexto social. Por este motivo es la propia sociedad quien promueve las experiencias más innovadoras, las cuales se sitúan fuera del sistema educativo como por ejemplo: el exitoso proyecto «The Intel Computer Clubhouse Network» (Valverde, Fernández & Garrido, 2015) o, a través de diferentes organizaciones como Codeacademy (<https://www.codeacademy.com/es>) o Code.org (<https://code.org>). Fomentando aprendizajes que permitan a los ciudadanos no sólo tener mejores aptitudes para el mercado laboral, sino también aprovechar las ventajas y beneficios que el pensamiento computacional aporta al desarrollo personal y social.

## 4. Justificación

La incorporación de estos conocimientos en la educación formal pretende que los individuos no sean meros consumidores de productos tecnológicos digitales y formen parte activa de la Sociedad Digital en todas sus dimensiones (Clark, Rogers, Spradling, & Pais, 2013; Gardner & Feng, 2010; Lambert & Guiffre, 2009).

La escuela está obligada, por un lado, a no volver la espalda a la realidad social de cada momento y, por otro, a tratar de capitalizar en beneficio de sus funciones las oportunidades que se le presenten. Dos de los más inmediatos retos del sistema educativo actual son: primero, responder adecuadamente a la demanda de alfabetización digital de los jóvenes en un entorno cada vez más multimedia y, segundo, incorporar en sus metodologías docentes las nuevas herramientas venidas de la mano del progreso tecnológico (González, 2009, p. 126).

Los modelos educativos deben comenzar a desarrollar, por tanto, un proceso de cambio, constituyendo como elementos de dicha renovación: el desarrollo de competencias, el pensamiento crítico y lógico, el trabajo cooperativo y colaborativo y el trabajo por proyectos de forma interdisciplinar.

Training and education of young people today should not be omitted from the process of ICT education. Application of technology in society has led to such a stage that ICT skills with the knowledge of reading, writing and numeracy considered starting an elementary literacy (Stošić & Bogdanović, 2013, p. 1).<sup>2</sup>

Esta situación provoca que, paulatinamente, se vayan generando entornos tecnológicos, o no, bien en centros públicos o concertados, bien en empresas privadas donde los discentes aprenden Ciencias, Matemáticas, Lengua, Inglés, Arte o Música teniendo el pensamiento computacional como base. Este aprendizaje lo pueden lograr a través de diferentes experiencias que tiene como herramientas desde el sencillo lápiz y papel, un novedoso software de programación visual, como Scratch o, incluso, a través de robótica educativa con LEGO, aplicado a la vida real a través del proyecto *LEGO Engienier*, donde la programación realizada en *Scratch* se traduce en acciones en el mundo físico.

---

<sup>2</sup> "La formación y la educación de los jóvenes de hoy no deben ser omitidas del proceso de enseñanza de las TIC. La aplicación de la tecnología en la sociedad ha conducido a un estadio tal que las competencias en TIC junto con el conocimiento de la lectura, la escritura y la aritmética son considerados el inicio de una alfabetización elemental." (Stošić y Bogdanović, 2013, p.1)

Las repercusiones del pensamiento computacional en todos los niveles de la Educación son innumerables, así como en el desarrollo cognitivo de los educandos que están expuestos a dicha actividad. Para Pea y Kurland (1984) su contribución a las habilidades matemáticas viene determinada por su relación con la informática. Asimismo, la actividad derivada del uso de la programación informática requiere de una continua utilización de la memoria y de una capacidad de concentración que permita controlar las diferentes variables que definen un problema, desarrollando ambas competencias. Además, la interconexión que puede establecerse entre la forma de resolver y la resolución de problemas en el entorno próximo del alumnado permite un incremento de las habilidades relacionadas con el razonamiento analógico, fomentando un uso interdisciplinar de las habilidades involucradas en dicho proceso.

Como consecuencia de la utilización continuada de sentencias condicionales, a través de las cuales se establece una relación causa-efecto entre dos variables, se favorece la adquisición de un razonamiento condicional motivado por las conexiones lógicas que se establecen. Supeditada a la programación informática resalta, también, el desarrollo de un pensamiento procedimental, como resultado de la redacción de instrucciones de diferentes dificultades para ser ejecutadas por el programa, fomentando la toma de decisiones a la hora de actuar en la vida real.

Debido a todas estas cuestiones surge la necesidad de realizar una revisión sistemática de la producción científica sobre el pensamiento computacional. Casos donde se realizan experiencias reales o estudios que tienen como objetivo potenciar este concepto en ámbitos formales, nos pueden aportar a la comunidad educativa una base desde la que comenzar o impulsar investigaciones sobre el desarrollo del pensamiento computacional en la educación.

Con la pretensión de establecer una revisión significativa de los estudios realizados será esencial tener en consideración una serie de competencias<sup>3</sup> adquiridas durante el Máster en Educación Digital que permitan un correcto desarrollo de la presente propuesta, así como la consecución de los objetivos establecidos para su conclusión.

---

3 Tabla 1

Tabla 1. Competencias necesarias para el desarrollo del Trabajo Fin de Máster.

Fuente: Elaboración propia a partir del VERIFICA del Máster en Educación Digital.

Competencias	
<b>Generales</b>	
CG1	Ser capaz de diseñar e implementar un estudio de investigación original y riguroso sobre un problema significativo (pedagógico, sociológico, cultural, económico, político o ético), relacionado con la tecnología educativa.
<b>Básicas</b>	
CB8	Que los estudiantes sean capaces de integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios.
CB9	Que los estudiantes sepan comunicar sus conclusiones y los conocimientos y razones últimas que las sustentan, a públicos especializados y no especializados de un modo claro y sin ambigüedades.
<b>Específicas</b>	
CE1	Desarrollar conocimientos avanzados en Tecnología Educativa y demostrar una comprensión fundamentada de la teoría y la práctica pedagógica en el ámbito de la Educación Digital.
<b>Transversales</b>	
CT2	Utilizar las nuevas tecnologías de la información como instrumento de trabajo intelectual y como elemento esencial para informarse, aprender y comunicarse.
CT4	Adquirir los conocimientos metodológicos necesarios para afrontar los retos profesionales o de investigación de una forma ética y rigurosa.

## **5. Objetivos**

Los objetivos, tanto generales como específicos, giran entorno a la realización de una revisión sistemática sobre el pensamiento computacional en todas las etapas de la educación. Para ello, partiremos de un conjunto de incógnitas cuyas respuestas nos permitirán obtener la base de conocimiento deseada y, por ende, los objetivos de la presente investigación.

¿Desde qué temas se aborda el pensamiento computacional en los últimos años? ¿A qué niveles académicos van dirigidas las investigaciones? ¿Qué metodologías utilizan para realizar los artículos? ¿Cómo proponen los últimos artículos incluir el pensamiento computacional en el currículum? ¿Cuántos estudios buscan desarrollar el pensamiento computacional en la educación?

### **5.1. Objetivo General**

- Realizar una revisión sistemática del concepto pensamiento computacional a través del análisis de casos, estudios y experiencias realizadas.

### **5.2. Objetivo Específico**

- Estudiar las diferentes temáticas desde las que se estudia el pensamiento computacional.
- Examinar los diferentes niveles académicos a los que van dirigidos los artículos.
- Analizar las metodologías seguidas en los diferentes proyectos.
- Examinar las propuestas de inclusión del concepto en el currículum por los diferentes artículos.
- Estudiar aquellas experiencias reales llevadas a cabo en espacios de educación formal.

## **6. Marco Teórico**

Con ánimo de alcanzar los objetivos propuestos en esta investigación documentaremos la evolución sufrida por el pensamiento computacional desde su propia definición, así como sus inicios en los primeros lenguajes de programación y su evolución en los últimos diez años hasta llegar a la intención de incluirlo en la educación formal.

### **6.1. Definición de pensamiento computacional.**

El concepto del pensamiento computacional es un término que mantiene un alto nivel de estereotipos, debido al pensamiento común de ser algo propio de un grupo concreto de personas. Ligado durante mucho tiempo al mundo de las ingenierías de la informática y la computación, desde hace una década comienzan a surgir corrientes que pretenden incorporar el pensamiento computacional en el currículum oficial educativo, a través de experiencias y proyectos vinculados a los lenguajes de programación visual o al desarrollo del concepto con un sencillo lápiz y una hoja de papel.

Un concepto que aplicado en las personas no sólo desarrollan habilidades basadas en la informática, sino que estarán en disposición de poder resolver problemas complejos al mismo tiempo que obtendrán la capacidad de entender ese lenguaje abstracto que es el lenguaje de los ordenadores. Unas habilidades que extrapoladas a problemas más cotidianos potenciarán la obtención de su solución. Mas no podemos definir el pensamiento computacional como una sencilla habilidad.

Para Wing (2006, p. 33) el «pensamiento computacional» consiste en «solving problems, designing systems and understanding of human behavior, using the fundamental concepts of computer science».<sup>4</sup> Un pensamiento basado en procesos realizados por un humano o una máquina con métodos y modelos que permiten resolver problemas y diseñar sistemas que por sí solos no podrían (Kafai & Burke, 2014; Sengupta, Kinnebrew, Basu, Biswas, & Clark, 2013).

---

<sup>4</sup> «Resolución de problemas, diseño de sistemas y la comprensión del comportamiento humano, el uso de los conceptos fundamentales de la informática.

Un enfoque para resolver un determinado problema que empodera la integración de tecnologías digitales con ideas humanas. No reemplaza el énfasis en creatividad, razonamiento o pensamiento crítico pero refuerza esas habilidades al tiempo que realza formas de organizar el problema de manera que el computador pueda ayudar (CSTA and ISTE, 2011).

Se trata de una competencia básica que todo ciudadano debería conocer para desenvolverse en la sociedad digital, pero no es una habilidad «mecánica», ya que es una forma de resolver problemas de manera inteligente e imaginativa. Los conceptos computacionales se utilizan para enfocar y resolver problemas reales, comunicarnos con otras personas y gestionar múltiples aspectos de nuestra vida cotidiana (Wing, 2006).

Asimismo, en el Massachusetts Institute of Technology el equipo de Scratch define el pensamiento computacional como una agrupación de conceptos, prácticas y perspectivas que se fundamenta en el mundo de la informática. Aprender programando y compartiendo sus creaciones provoca en los discentes que se desarrollen como pensadores computacionales: se aprenden conceptos básicos al mismo tiempo que se desarrollan estrategias de diseño, resolución de problemas y maneras de colaborar (ScratchEd Team, 2015).



Ilustración 1: Palabras usadas común mente para definir el pensamiento computacional.

Fuente: Recuperado de (Kalelioğlu, Gülbahar, & Kukul, 2016)

Asimismo, el pensamiento computacional es visto como una metodología que implementa conceptos básicos de la computación para resolver todo tipo de problemas, diseñar estrategias y realizar tareas. Una forma de poder abordar los problemas que nos ofrece eficacia y posibilidades de éxito que de otra forma no sería posible. (Olabe, Basogain, & Basogain, 2015)

Now that the student can program Space Invaders, can the student program a science simulation? (Han, Basawapatna, Bennett & Repenning, 2010)<sup>5</sup>

La autora con mayor relevancia en el estudio del pensamiento computacional Jeannette Wing, propone una forma de resolver los problemas a través del pensamiento computacional, ya sea con el uso de herramientas digitales o no. Wing expone una serie de características y propiedades que son desarrollados con el uso del pensamiento computacional. En la Tabla 2, recuperada de Olabe, et al. (2015), se puede observar de forma resumida los principales conceptos que se asocian a la propuesta de Wing, unas características que dan respuesta a la pregunta realizada por Han, et al. (2010).

Por tanto, las repercusiones de la programación informática en todos los niveles de la Educación son innumerables, así como en el desarrollo cognitivo de los educandos que están expuestos a dicha actividad. Para Pea y Kurland (1984) su contribución a las habilidades matemáticas viene determinada por su relación con la informática. Asimismo, la actividad derivada del uso de la programación informática requiere de una continua utilización de la memoria y de una capacidad de concentración que permita controlar las diferentes variables que definen un programa, desarrollando ambas competencias. Además, la interconexión que puede establecerse entre la forma de programar y la resolución de problemas en el entorno próximo del alumnado permite un incremento de las habilidades relacionadas con el razonamiento analógico, fomentando un uso interdisciplinar de las habilidades involucradas en dicho proceso.

---

<sup>5</sup> Ahora que los estudiantes pueden programar un Space Invaders, ¿puede un estudiante programar una simulación científica? (Han, Basawapatna, Bennett & Repenning, 2010)

Tabla 2. Conceptos y características del pensamiento computacional.

Fuente: Olabe, et al., 2015.

Conceptos y Características	
Nº	Característica
1	Reformular un problema a uno parecido que sepamos resolver por reducción, encuadrarlo, transformar o simular.
2	Pensar recursivamente.
3	Procesar en paralelo.
4	Interpretar código como datos y datos como código.
5	Generalizar análisis dimensional.
6	Reconocer ventajas y desventajas del solapamiento.
7	Reconocer coste y potencia de tratamiento indirecto y llamada a proceso.
8	Juzgar un programa por simplicidad de diseño.
9	Utilizar abstracción y descomposición en un problema complejo o diseño de sistemas complejos.
10	Elegir una correcta representación o modelo para hacer tratable el problema.
11	Seguridad en utilizarlo, modificarlo en un problema complejo sin conocer cada detalle.
12	Modularizar ante múltiples usuarios.
13	Prefetching y caching anticipadamente para el futuro.
14	Prevención, protección, recuperarse de escenario peor caso.
15	Utilizar razonamiento heurístico para encontrar la solución.
16	Planificar y aprender en presencia de incertidumbre.
17	Buscar, buscar y buscar más.
18	Utilizar muchos datos para acelerar la computación.
19	Límite tiempo/espacio y memoria/potencia de procesado.

En definitiva, un proceso de aprendizaje que se diferencia de las metodologías tradicionales y que, sin embargo, se podría desarrollar en la mayoría de actividades comunes que se realizan en las aulas. En palabras de Olabe et al. (2015) si los problemas planteados por el Programa Internacional para la evaluación de estudiantes (PISA) se desarrollasen con el uso del pensamiento computacional, concretamente con la herramienta digital Scratch., se lograría no sólo resolver el problema más inmediato sino favorecer el desarrollo de las ya mencionadas competencias y habilidades.

## **6.2. Evolución de su aplicación en la educación.**

Llegado el momento, el ser humano tuvo la imperiosa necesidad de comunicarse con la máquina (Balagurusamy, 2010). Fue entonces cuando los primeros lenguajes de programación comenzaron a evolucionar, códigos y símbolos de alto nivel de complejidad, cuando el pensamiento computacional vio la luz.

Aunque el término se encuentra en un estado de madurez temprana, existen más de cincuenta años de literatura que tienen su comienzo en los años sesenta, cuando un matemático llamado Seymour Papert, discípulo de Jean Piaget, co-fundó el Massachusetts Institute of Technology (MIT) Artificial Intelligence Laboratory con su compañero Marvin Minsky. Papert desarrolló en 1967 la primera versión de un lenguaje de programación visual: Logo. Así, si en los últimos diez años el foco del pensamiento computacional se ha puesto sobre Jannette Wing's, Papert fue el pionero de la idea de que los niños pudieran desarrollarlo a través de la programación con Logo.

El entorno más conocido fue «Turtle», una simpática tortuga que podía moverse a voluntad del usuario sobre una superficie. Cada movimiento era recogido e interpretado como un recorrido. Finalmente, el itinerario terminaba siendo una forma definida. Este proceso era desarrollado únicamente con dos órdenes: «forward» and «right».

Esa tortuga había sido confeccionada para superar las barreras que el lenguaje de programación tradicional, incluidos los lenguajes más sencillos diseñados para comenzar en el mundo de la programación, presentaba a los usuarios Resnick et. al. (2009). Tres serían, por tanto, las ideas su sintaxis debía ser sencilla adaptada al discente, las actividades debían estar conectadas con sus intereses y experiencias y, por último, debía permanecer en un contexto cercano al estudiante.

El recorrido vio su esplendor con la publicación de Seymour Papert de «Mindstorms: Children, Computers, and Powerful Ideas» (1980), un libro que fascinó con el potencial del que disponía Logo. Más tarde, LEGO Logo llegaría de la mano de Mitchel Resnick que desarrolló un sistema para conectar el lenguaje de programación que el usuario desarrollaba, a través del software, con el mundo real. Motores, luces o sencillos sensores podían dar vida a construcciones realizadas con los bloques de LEGO y otros elementos. Hasta llegar a las aproximadamente más de trescientas versiones de Logo que han tenido una vida limitada o que continúan evolucionando en la actualidad. En definitiva, había nacido un nuevo entorno de aprendizaje.

Más tarde, LEGO Logo innovaría en el mercado con su Programmable Brick, un bloque en el que se podía descargar el programa creado por el usuario y al ser conectado con el motor realizaba su programación base de forma autónoma. Este “ladrillo programable” ha tenido varias versiones: en sus inicios fue el RCX, más tarde pasó a llamarse NXT y en la actualidad se le conoce por EV3, perteneciente a la gama de productos LEGO Mindstorms.

Al comienzo de la década de los noventa, apareció una nueva versión denominada «MicroWorlds», creada por Logo Computer Systems Inc. (LCSI). Aunque la versión incluía nuevas extensiones de dibujo, editores de forma, melodía y la posibilidad de importar gráficos y sonidos, el verdadero avance de esta versión fue la posibilidad de multitarea o procesamiento paralelo. Poder realizar varios procesos al mismo tiempo era posible en Logo, pero de una forma muy compleja frente a la naturalidad de MicroWorlds.

Tras ello, se suceden una serie de versiones que mantienen la idea inicial de Seymour Papert viva, aunque el proyecto inicial dejara de evolucionar. Aproximadamente, son más de trescientas las versiones de Logo que han tenido una vida limitada o que continúan en la actualidad.

Países latinoamericanos, Japón o Austria fueron algunos de los países que tuvieron proyectos educativos basados en Logo. Es en este contexto donde comienza a germinar la idea de incluir el pensamiento computacional dentro del aula. A través de herramientas educativas, ya sea como una asignatura más, bien como un contenido transversal, como un proyecto educativo o, sencillamente, como un complemento didáctico más. Aunque la corriente actual apuesta por una competencia básica del siglo XXI, que no debería estar limitada dentro de unos límites concretos sean los de una asignatura o del propio currículo (Goode, Margolis, & Chapman, 2014).

### **6.3. El pensamiento computacional en los últimos diez años.**

“It represents a universally applicable attitude and skill set everyone, not just computer scientists would be eager to learn and use.” (Wing, 2006, p.1). Con estas palabras comenzaba hace diez años Jeannette Wing's un influenciante artículo, cuyos argumentos captaron la atención de la comunidad educativa e impulsarían las corrientes mencionadas.

Los investigadores y educadores actualmente trabajan, en su mayoría con la abstracción como piedra angular del pensamiento. Entendiendo la abstracción como la definición de patrones y la generalización a partir de casos concretos, la clave para tratar la complejidad (Wing, 2011). Un concepto que ha llevado a los siguientes elementos a ser aceptados e incluidos en diferentes planes de estudios que tienen como objetivo desarrollar y evaluar el pensamiento computacional (Grover & Pea, 2013):

- Abstracciones y generalizaciones de patrones.
- Procesamiento sistemático de información.
- Sistemas de símbolos y representaciones.
- Nociones de algoritmos de flujo de control.
- Descomposición estructurada de problemas.
- Pensar con lógica condicional.
- Limitaciones de eficiencia y rendimiento.
- Depuración y detección de errores sistemáticos.

Además, encontraremos que se encuentran bajo investigación la idea del pensamiento computacional como medio para enseñar otras asignaturas. Desde que se usara Logo para enseñar matemáticas o ciencias, se ha ligado el pensamiento computacional al denominado STEM<sup>6</sup> (Science, Technology, Engineering and Mathematics) y por extensión el desarrollo del idioma inglés al ser el idioma de los comandos. Sin embargo, comienzan a introducirse las habilidades de resolución de problemas en el resto de materias (Grover & Pea, 2013).

Y así, el pensamiento computacional y la programación empiezan a formar parte del currículo oficial en los sistemas educativos formales (Záhorec, Hašková, Munk, 2014). En definitiva, una curva de crecimiento exponencial que se aprecia en el gran número de países que incorpora el pensamiento computacional o el «coding» entre sus planes de estudio, cifra que sigue en aumento.

---

6 CTIM (Ciencia, Tecnología, Ingeniería y Matemáticas).

Países como Rusia, Sudáfrica, Nueva Zelanda y Australia han realizado cambios en sus currículums oficiales de educación primaria. Reino Unido inició en 2012 programas pilotos para llevar la programación a todos los niños y niñas, para más tarde implementar la programación informática en sus escuelas de Primaria y Secundaria, al mismo tiempo que Estonia lo introducía en el currículum. Mientras el sistema educativo finlandés promueve un conocimiento comprensivo que permita al estudiante construir y comprender las TD para ser capaces de crear sus propias aplicaciones. O en nuestro propio país, siguiendo el modelo británico, la Consejería de Educación del Gobierno de la Comunidad Autónoma de Madrid cuando incluyó la asignatura de libre configuración autonómica denominada «Tecnología, Programación y Robótica». Implicando, así, un enaltecimiento de la importancia de adaptar el currículum para enseñar un pensamiento lógico para crear y resolver programas. Una dimensión tangible donde el pensamiento computacional y la programación introducen la materialidad y la realidad física.

Los sistemas educativos están cada vez más obligados a adaptarse a la realidad social y dar respuesta a las demandas de alfabetización digital y la incorporación de metodologías docentes innovadoras, que no necesariamente deben serlo por el uso de nuevas herramientas llegadas con el progreso tecnológico, sino por un cambio más profundo que incluya la incorporación de competencias como el pensamiento computacional (González, 2009).

Se trata de diseñar una nueva ecología de aprendizaje que permita a los estudiantes comprender cómo funciona el mundo al mismo tiempo que les permita aprender a resolver problemas complejos (Johnson, Adams, Estrada & Freeman, 2014). Y es que la actual tendencia apunta hacia la demanda y necesidad de los discentes por conocer cómo funcionan las Tecnologías Digitales para aprovechar todas sus potencialidades.

Una demanda que se ve reconocida con el volumen de herramientas digitales que están a disposición de los usuarios. Pues siguiendo la línea de Scratch en cuanto a programación visual podemos encontrar decenas de programas como los mostrados en la Tabla 3.

Tabla 3. Relación de programas para desarrollar la programación de forma visual.

Fuente: Elaboración propia.

Software de Programación Visual	
Scratch	Alice
Kodu	Game Maker
Greenfoot	Toontalk
Logo	Stagecast Creator
Swift Playgrounds	

Alice, imitando al original de Lewis Carroll, permite programar en entornos de tres dimensiones, comunidades como Game Maker y Kodu, donde la creación de videojuegos es una realidad, Greenfoot, donde aprender a programar en Java de forma visual, o Swift Playgrounds, la última aportación de Apple al mundo educativo, sin olvidar las páginas web que simulan entornos de programación o los equipos de robótica que permiten unir el mundo digital y el físico.

El aprendizaje es un proceso social y, especialmente en el ámbito de la cultura digital, las comunidades de usuarios conforman una nueva ecología de aprendizaje caracterizada por una alta motivación hacia la participación y la colaboración. Los actuales entornos de aprendizaje más complejos, distribuidos, conectados y flexibles, exigen incorporar una nueva metáfora sobre la «creación del conocimiento», es decir, la creación colectiva del conocimiento por medio de objetos de actividad compartidos. Este enfoque se conoce como «aprendizaje trialógico» (co-evolución de investigadores, de objetos de investigación).

Los objetos pueden ser conceptuales (preguntas, teorías, diseños) o materiales (prototipos, productos concretos) o representar prácticas sobre las que reflexionar o transformar colectivamente. El aprendizaje es un proceso de indagación innovadora en el que el objetivo es refinar progresivamente los artefactos de conocimiento y desarrollar procesos a largo plazo para extender el conocimiento de la comunidad y sus competencias (Hakkarainen, 2009).

La creación de programas no es un proceso que se realice de modo aislado sino en un contexto social. Las experiencias más innovadoras se sitúan fuera del sistema educativo como, por ejemplo, el exitoso proyecto «The Intel Computer Clubhouse Network» (Valverde, et al. 2015). En efecto, la sociedad civil, a través de diferentes organizaciones como Codecademy o Code.org, está implicada en el fomento de aprendizajes que permitan a los ciudadanos no sólo tener mejores aptitudes para el mercado laboral, sino también aprovechar las ventajas y beneficios que el pensamiento computacional aporta al desarrollo personal y social. La incorporación de estos conocimientos en la educación formal pretende que los individuos no sean meros consumidores de productos tecnológicos digitales y formen parte activa de la Sociedad Digital en todas sus dimensiones (Clark, Rogers, Spraling & Pais, 2013).

## **7. Metodología**

En la última década la gran producción de artículos de investigación y la globalización y accesibilidad del conocimiento está generando problemas en la actualización del individuo, que debería leer decenas de artículos diariamente para poder conocer las últimas publicaciones de un tema concreto.

Una revisión sistemática sigue una metodología que nos permite interpretar y evaluar toda la información accesible y relevante sobre una cuestión determinada. Por ello, las razones para llevar a cabo una revisión sistemática son prácticas, necesitamos de este tipo de investigación para tener bases racionales sobre las que seguir investigando (Manterola, Astudillo, Arias, & Claros, 2013).

Definida como una metodología sistemática y flexible cuyo fin es el de mejorar las prácticas a través de la búsqueda, análisis y obtención de resultados. Siempre desde una postura integradora, el presente enfoque, además de presentar una alternativa de estudio, nos permite obtener como resultado un nuevo o contrastado conocimiento de las prácticas realizadas en los últimos años.

Por tanto, para su correcto desarrollo como investigación deberemos tener en cuenta un conjunto de fases. A saber: formulación del problema, búsqueda de las fuentes de estudios primarios, codificación de las fuentes, análisis de los resultados y síntesis general, unas fases que nos permitirán mantener una metodología objetiva con la que realizar el proceso de revisión y obtener un conjunto de evidencias liberadas de toda subjetividad (Sánchez, 2010).

### **7.1. Formulación del problema.**

Toda investigación, indistintamente del tipo seleccionado, requiere de un primer paso inicial: definición del problema. En la presente revisión sistemática se identificó el incremento de publicaciones sobre el pensamiento computacional en los últimos años. Un hecho que potencia la necesidad de revisiones con las que mantener el estado de la cuestión actualizado.

Para todo problema debemos plantearnos una serie de cuestiones a las que contestar, que además nos servirán para acotar nuestra búsqueda en la investigación. Tal y como hemos observado en los objetivos del presente texto las preguntas a las que daremos respuesta están ya predefinidas, con lo cual los temas a los que debemos dar respuesta con nuestros resultados serán:

- Análisis de la finalidad de los artículos.
- Análisis del rango académico al que van dirigidos los artículos.
- Metodologías o tipos de artículos encontrados.
- Propuestas de introducción del pensamiento computacional en el currículum.
- Experiencias reales en la educación formal.

Para ello, comenzaremos nuestra búsqueda por localizar, en caso de existir, las últimas revisiones sistemáticas sobre el concepto. Comprobaremos si los mismos solucionan nuestras cuestiones iniciales para en caso afirmativo continuar, mejorar o actualizar su trabajo realizando una búsqueda más actual o ampliando los campos de búsqueda utilizados y en caso negativo realizar una búsqueda más profunda sobre el tema concreto.

## **7.2. Búsqueda y localización de las fuentes de estudios primarios.**

Antes de comenzar a investigar debemos seleccionar las bases de datos que usaremos para realizar nuestras búsquedas. Aunque podemos hacer uso de un número infinito de buscadores es recomendable hacer uso de bases de datos especializadas en la temática, en esta ocasión con educación, como puede ser ERIC u otras bases multidisciplinares como Dialnet, Web of Science, etc.<sup>7</sup> Asimismo, la presente investigación se basa en artículos publicados en revistas científicas, por lo que debemos tener especial atención en no hacer uso de repositorios.

---

<sup>7</sup> Debemos recordar tener cuidado con las duplicidades.

Tabla 4. Bases de datos seleccionadas. Fuente: Elaboración propia.

Bases de Datos
ACM Digital Library
Science Direct
ERIC
Dialnet
Web of Science

Una vez seleccionadas las bases de datos nuestro siguiente paso debe ser la definición de los criterios de búsqueda que vamos a seguir. Estos deben estar claros para ser usados en todos los buscadores por igual, si deseamos recabar resultados objetivos y correctamente obtenidos. En caso de que algún criterio no pueda ser especificado dentro de la búsqueda, debido a las especificaciones técnicas del portal web, será notificado en los propios resultados.

Tabla 5. Criterios de búsqueda. Fuente: Elaboración propia.

Nº	Criterio
C1	Campo Abstract o Resumen.
C2	Palabra clave Computational Thinking o Pensamiento Computacional.
C3	Fecha de publicación Desde 2014.
C4	Tipo Artículo de Revista Científica.

A continuación se detallarán los criterios con los que se determinarán qué artículos estarán incluidos en la revisión y, por ende, cuales no:

C1. La búsqueda en el abstract o resumen nos permitirá tener acceso a un mayor volumen de artículos que contengan el concepto como parte importante del mismo. Restringirlo al título o a las palabras clave reduciría nuestro campo de acción y suprimiría artículos por no incluir el concepto en campos muy limitados.

C2. El uso de la palabra clave en castellano e inglés nos amplía el campo de búsqueda a nivel mundial sin olvidar la producción hispano hablante. De nuevo, utilizar más palabras clave nos seccionaría artículos que no usaran esos conceptos concretos.

C3. Aunque en las revisiones sistemáticas suelen fijar un rango de fecha de publicación más amplio, de unos 5 años, en esta ocasión con motivo de las últimas revisiones sistemáticas realizadas sobre la temática hemos reducido el campo a los años 2014, 2015 y 2016.

### 7.3. Codificación de las fuentes.

Las revisiones sistemáticas han visto aumentado su uso en los últimos años, por los motivos anteriormente comentados. Sin embargo, no todas poseen una calidad aceptable debido a que no son sistemáticas, no siguen una metodología predefinida o presentan sesgos de diferentes tipos (Manterola, et al. 2013).

Por todo ello, además de fijar los criterios de búsqueda para todas las bases de datos, debemos especificar cuáles serán las razones de exclusión (RE) para descartar un artículo de la selección final. En esta ocasión hemos estimado oportuno seleccionar los indicados en la tabla número seis.

Tabla 6. Razones de Exclusión. Fuente: Elaboración propia.

Nº		Razón
RE1	Idioma	Únicamente se aceptarán en castellano e inglés.
RE2	Temática	Deberán pertenecer a educación y al desarrollo del concepto.
RE3	Artículo	Aún no se ha publicado o pendiente de difusión.

Estas serán las únicas razones por las que un artículo quedaría excluido de la revisión y seguirlas nos permitirán obtener unos resultados sistemáticos y objetivos.

RE1. En caso de que el artículo esté publicado en otro idioma será excluido para no realizar una interpretación errónea del mismo que pudiera enturbiar los resultados.

RE2. El pensamiento computacional puede ser tratado en diferentes disciplinas que quedarían fuera de la revisión por no tratarse del tema de investigación. Así mismo ocurrirá con aquellos artículos que traten el concepto de forma superficial.

RE3. En ocasiones se publican los resúmenes o abstract de futuros artículos que aún no están disponibles a la comunidad y, por tanto, serán excluidos con el objetivo de no interpretarlos erróneamente.

#### **7.4. Análisis de los resultados.**

Tras seleccionar de forma deductiva, cumpliendo con los criterios y razones de exclusión, realizaremos una tabla con los resultados obtenidos en cada base de datos, donde encontraremos los localizados, los excluidos y los seleccionados. Así mismo, se realizará una plantilla con la información principal de los artículos seleccionados que nos permita dar respuesta a los interrogantes iniciales. Por tanto, incluirá la temática específica, el rango académico al que va dirigido, la metodología seguida, si contiene propuestas de inclusión en el curriculum o no y si es una experiencia o no.

Para la realización de dicha plantilla, partiremos de las palabras clave y la lectura de los resúmenes o abstracts. Aplicaremos la codificación, anteriormente explicada, lo cual permitirá seleccionar aquellos documentos que traten directamente sobre los objetivos ya marcados. Este proceso deberá ser realizado de forma cautelosa, debido a la heterogeneidad de los artículos. Siguiendo de forma precisa la metodología marcada, para así evitar el problema de realizar una interpretación de los resultados errónea (Manterola, et al.2013).

#### **7.5. Síntesis general o conclusiones.**

Una vez expuestos los resultados obtenidos, tras la realización de un análisis objetivo, para las diferentes cuestiones, deberemos realizar una síntesis final o conclusión que permita recoger lo descubierto por nuestra investigación. Aunque los resultados por sí mismos, acompañados por sus gráficas y tablas correspondientes, pondrán al lector en la situación actual del tema, la síntesis generará una base sobre la que continuar investigando o realizando nuevas revisiones sistemáticas en el futuro.

## **8. Resultados**

Los datos obtenidos implican que se realice una lectura profunda de los artículos científicos publicados durante los últimos años para la realización de un análisis, de aquellos seleccionados, que pueda dar respuesta a los objetivos planteados y así especificar las finalidades de los artículos, los rangos académicos a los que se dirigen, las metodologías usadas para alcanzar sus objetivos, indicar si son propuestas curriculares y determinar si son experiencias educativas reales que tienen como base desarrollar el pensamiento computacional.

Queda, por tanto, realizar un análisis de la información recabada no sólo por los seis buscadores seleccionados sino también de los datos recogidos de las revisiones ya realizadas. Para ello, el análisis planteado se mostrará desarrollado en diferentes tablas con varios apartados, que garanticen un estudio detallado y objetivo de los resultados obtenidos en ambas partes y permitan la interpretación de los mismos.

Sin embargo, como ya mencionamos en el marco metodológico, antes de realizar la búsqueda debemos tener en cuenta que se han realizado, al menos, tres revisiones en los últimos años. En el caso de Grover & Pea (2013) nos muestra qué es y por qué es importante el pensamiento computacional, un resumen de las últimas herramientas digitales y un breve resumen del estado actual del concepto en la educación primaria. Mientras en Lye & Koh (2014) se realiza una revisión sistemática desde el año 2009 a el año 2013. Asimismo, en Kaleliñoğlu, Gülbahar & Kukul (2016) se realiza también una revisión sistemática, mas en esta ocasión entre los años 2006 y 2014.

Esta base marcará la presente revisión sistemática que buscará continuar el trabajo ya realizado y concretar cuestiones que los autores anteriores no reflejaron en sus trabajos. Es por esta razón, que la búsqueda realizada tendrá lugar entre los años 2014 y agosto de 2016, un conjunto de tres años que permitirá saber el estado actual de las cuestiones de cara a futuras investigaciones. Asimismo, la configuración de los temas determinados estarán relacionados, en gran medida, con los ya tratados en sus publicaciones.

Tabla 7. Resultados obtenidos de la búsqueda. Fuente: Elaboración propia.

Portal Web	Total	RE1	RE2	RE3	Seleccionados	Notas
ACM Digital Library	58	0	43	10	4 (5) <sup>8</sup>	
Science Direct	27	0	23	0	4	Title & Keyword
ERIC	51	0	34	1	3 (16)	
Dialnet	16	2	8	0	6	
Web of Science	32	0	4	1	27	Topic
Total	185	2	112	12	45 (59)	

Conforme a lo dictado en la metodología y las revisiones ya existentes, se realizó la búsqueda. Obteniendo como resultados los mostrados en la Tabla 7. Para la codificación de los resultados obtenidos se realizaba una lectura de cada uno de los artículos, lo cual amplió o redujo la búsqueda intentando, siempre, evitar la duplicidad de estudios sobre la temática.

Intentando lograr la mayor selección posible la búsqueda se limitó a usar como palabra clave pensamiento computacional. Esta decisión aumentó el número de RE2 obtenidos debido a que no tenían que ver con el tema central de esta revisión. Este apartado se podría haber reducido utilizando un mayor número de términos, pero se hubieran limitado artículos que por otras razones no fueran tan específicos.

En cuanto a los datos RE1 y RE3 son meramente anecdóticos y se dan en bases de datos que no permiten afinar correctamente la búsqueda. En el caso de RE3 hay que matizar que la búsqueda se realizó a finales del mes de agosto y, por tanto, cualquier publicación posterior del mes de septiembre quedaría completamente descartada. En las notas únicamente se refleja que dos de las bases de datos no permiten la búsqueda en el abstract o resumen como si permitía el resto. Finalmente de los 185 artículos científicos encontrados fueron seleccionados 45, los cuales son expuestos en la Tabla 8.

<sup>8</sup> (x) Artículos repetidos. Se incluye para la correcta suma de los totales.

Tabla 8. Artículos seleccionados. Fuente: Elaboración propia.

Código	Título	Primer Autor	Fecha	Idioma	Tema	Rango	Metodología	Propuesta	Experiencia
Ar01	A Critical Look at the Digital Technologies in Architectural Education: When, where, and how?	Kara, Levent	2015	In	Análisis	-	Teoría	No	No
Ar02	A K-6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge	Angeli, Charoula	2016	In	Marco	E Pri	Teoría	Si	No
Ar03	Alexander Meets Michotte: A Simulation Tool Based on Pattern Programming and Phenomenology	Basawapatna, Ashok	2016	In	Método	E Pri	Cuantitativa	No	Si
Ar04	An Exploration of Three-Dimensional Integrated Assessment for Computational Thinking	Zhong, Baichang	2016	In	Integración	E Pri	Mixto	Si	Si
Ar05	Assessing the computational literacy of elementary students on a national level in Korea	Jun, SooJin	2014	In	Análisis	E Pri	Cuantitativa	No	No
Ar06	Bringing Computational Thinking to STEM Education	Swaid, Samar I.	2015	In	Análisis	E Sup	Revisión	No	No
Ar07	Characteristics of Computational Thinking about the Estimation of the Students in Mathematics ...	Promraksa, Siwarak	2014	In	Análisis	E Pri	Cualitativo	No	No
Ar08	Children Programming Games: A Strategy for Measuring Computational Learning	Werner, Linda	2015	In	Marco	E Pri	Cualitativo	No	Si
Ar09	Children's Perceptions of What Counts As a Programming Language	Lewis, Colleen	2014	In	Método	E Pri	Mixto	No	Si
Ar10	Comparing Virtual and Physical Robotics Environments for Supporting Complex Systems and ...	Berland, Matthew	2015	In	Análisis	E Pri	Mixto	No	No
Ar11	Computational Science for Undergraduate Biologists via QUT.Bio.Excel	Buckingham, Lawrence	2014	In	Integración	E Sup	Teoría	Si	No
Ar12	Computational thinking and tinkering: Exploration of an early childhood robotics curriculum	Bers, Marina Umaschi	2014	In	Integración	E Inf	Mixto	Si	No
Ar13	Computational Thinking as a Computer Science Education Framework and the Related Effects ...	Jon Good	2016	In	Análisis	E Sec	Cualitativo	No	No
Ar14	Computational Thinking Concepts for Grade School	Sanford, John F.	2016	In	Análisis	E Pri	Teoría	No	No
Ar15	Computational Thinking in Compulsory Education: Towards an Agenda for Research and Practice	Voogt, Joke	2015	In	Integración	-	Revisión	No	No
Ar16	Computational Thinking in Elementary and Secondary Teacher Education	Yadav, Aman	2014	In	Análisis	E Sup	Cuantitativa	No	Si
Ar17	Computational Thinking: Expanding the Toolkit	Imberman, Susan P.	2014	In	Método	Doc	Cualitativo	No	No
Ar18	Construction, categorization, and consensus: student generated computational artifacts as a ...	Wilkerson-Jerde, Michelle Hoda	2014	In	Análisis	E Pri	Cualitativo	No	Si
Ar19	Constructionist Gaming: Understanding the Benefits of Making Games for Learning	Kafai, Yasmin B.	2015	In	Análisis	-	Teoría	No	No
Ar20	Creative and Computational Thinking in the Context of New Literacies: Working with Teachers ...	DeSchryver, Michael D.	2015	In	Análisis	E Sup	Cualitativo	No	No
Ar21	Defining Computational Thinking for Mathematics and Science Classrooms	Weintrop, David	2016	In	Análisis	-	Teoría	No	No
Ar22	Democratizing Children's Computation: Learning Computational Science as Aesthetic Experience	Farris, Amy Voss	2016	In	Método	E Pri	Cualitativo	No	Si
Ar23	Determining the effects of computer science education at the secondary level on STEM major ...	Lee, Ahlam	2015	In	Análisis	E Sup	Cuantitativa	No	No

Código	Título	Primer Autor	Fecha	Idioma	Tema	Rango	Metodología	Propuesta	Experiencia
Ar24	El pensamiento computacional y las nuevas ecologías del aprendizaje	Valverde Berrocoso, Jesús	2015	Es	Integración	E Sec	Teoría	Si	No
Ar25	Empowering K-12 Students with Disabilities to Learn Computational Thinking and ...	Israel, Maya	2015	In	Método	E Pri	Teoría	No	No
Ar26	Evolución de la enseñanza de la informática y las TIC en la Escuela Media en Argentina ...	Cotik, Viviana	2016	Es	Análisis	-	Mixto	No	No
Ar27	Exploring Issues about Computational Thinking in Higher Education	Czerkowski, Betul C.	2015	In	Análisis	E Sup	Revisión	No	No
Ar28	Exploring Media Literacy and Computational Thinking: A Game Maker Curriculum Study	Jenson, Jennifer	2016	In	Análisis	E Pri	Cuantitativa	No	Si
Ar29	Exploring students' computational practice, design and performance of problem-solving ...	Chao, Po-Yao	2016	In	Análisis	E Sup	Mixto	No	No
Ar30	Gaming in Second Life via Scratch4SL: Engaging High School Students in ...	Pellas, Nikolaos	2016	In	Método	E Sup	Mixto	No	Si
Ar31	How to Teach Programming Indirectly--Using Spreadsheet Application	Tahy, Zsuzsanna Szalayné	2016	In	Análisis	-	Teoría	No	No
Ar32	Incorporación de la programación informática en el curriculum de biología	Rodríguez, Antonio Carvajal	2015	Es	Integración	E Sup	Teoría	Si	No
Ar33	Instructional supports for students with disabilities in K-5 computing: Findings from ...	Snodgrass, Melinda R.	2016	In	Análisis	E Pri	Cualitativo	No	Si
Ar34	Learning to communicate computationally with Flip: A bi-modal programming language ...	Howland, Kate	2015	In	Método	E Pri	Cualitativo	No	Si
Ar35	Más allá de las pantallas: experiencias en diseño y programación de objetos interactivos digitales	Bordignon, Fernando	2016	Es	Método	E Sec	Cualitativo	No	Si
Ar36	Pensamiento computacional y alfabetización digital	Zapata-Ros, Miguel	2015	Es	Análisis	-	Teoría	No	No
Ar37	Scalable Game Design: A Strategy to Bring Systemic Computer Science Education to Schools ...	Repenning, Alexander	2015	In	Método	E Pri	Teoría	No	No
Ar38	Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis	Israel, Maya	2015	In	Análisis	E Pri	Cualitativo	No	No
Ar39	Taking Computer Science and Programming into Schools: The Glyndŵr/BCS Turing Project	Grout, Vic	2014	In	Análisis	E Sup	Mixto	No	No
Ar40	Teachers' perception on Computational Thinking in Science Practices	Park, Se Young	2015	In	Análisis	Doc	Teoría	No	No
Ar41	Teaching Programming with Computational and Informational Thinking	Michaelson, Greg	2015	In	Análisis	-	Teoría	No	No
Ar42	Testing Algorithmic Skills in Traditional and Non-Traditional Programming Environments	Csernoch, Maria	2015	In	Análisis	E Sup	Mixto	No	No
Ar43	Una experiencia en la enseñanza de la programación para la permanencia de los alumnos ...	Arellano, Norma	2014	Es	Método	E Sup	Mixto	No	Si
Ar44	Using Games to Help Novices Embrace Programming: From Elementary to Higher Education	Mladenovic, Sasa	2016	In	Método	E Pri	Mixto	No	No
Ar45	Visual programming languages integrated across the curriculum in elementary school: A two ...	Saez-Lopez, Jose-Manuel	2016	In	Análisis	E Pri	Mixto	No	No

La presente tabla es una muestra-resumen que no sólo recoge los artículos seleccionados si no que además permite de forma visual comprobar los resultados obtenidos de la lectura de los mismos. En el detalle podemos encontrar el título del artículo, el primer autor, la fecha de publicación, el idioma en el que está escrito, la finalidad del mismo, el menor rango académico hacia el que va dirigido<sup>9</sup>, la metodología seguida por sus autores y si presentan propuestas o críticas de intervención al curriculum o experiencias reales en las aulas.

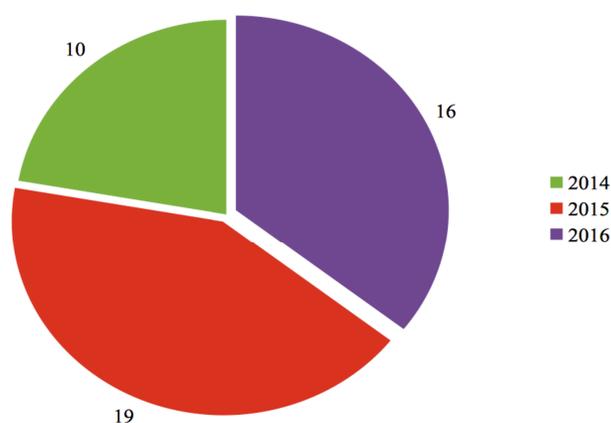


Gráfico 1. Número de artículos seleccionados por año de publicación. Fuente: Elaboración propia.

Uno de las referencias más curiosas al analizar el grueso de los datos obtenidos es con la prima variable: la fecha de publicación. Podemos observar en el Gráfico 1 como cada año aumenta la producción científica sobre el tema, pues aunque en el año 2015 supera al presente año, debemos recordar que la visión abarca hasta agosto y, por tanto, la tendencia, de continuar, será la de superar al año pasado. Muestra clara de que el pensamiento computacional se encuentra en auge, al menos para la comunidad educativa y sus investigadores.

<sup>9</sup> En algunos casos comienza en un nivel inferior y continuar con un nivel superior.

Los cuales se van especializando dentro de los diferentes campos, como ocurre con Maya, Tania y Jaime N. en Art25 y Art38 cuyos artículos están centrados en la educación primaria o en el caso de Joke Art2 y Art15 que se centran en aspectos más teóricos del concepto. Ejemplos de algunos de los autores que realizan un estudio continuo en el tiempo realizando aportaciones con sus investigaciones de forma constante.

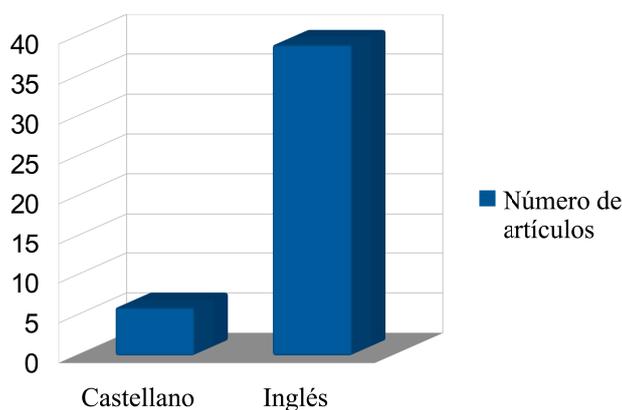


Gráfico 2. Idiomas en los que están redactados los artículos.

Fuente: Elaboración propia.

Asimismo, la segunda variable representada en el Gráfico 2 nos describe como el 77% de la producción científica se realiza en el idioma internacional. Mientras que tan sólo 6 artículos son redactados en castellano, de los cuales la gran mayoría provienen de países hispanohablantes.

### **8.1. Análisis de la finalidad de los artículos.**

Partiendo de las divisiones propuestas por Kalelíoğlu, et. al (2016) para el presente tema y teniendo en cuenta la similitud o igualdad entre algunas de las propuestas se ha decidido realizar una distribución más general en base a la finalidad principal que poseen las referencias seleccionadas. Por ello, procedemos a dividir el tema en cuatro focos posibles:

- Métodos o actividades que promueven el pensamiento computacional.
- Propuestas o análisis de la integración del pensamiento computacional en el currículum.
- Análisis, críticas o definiciones del pensamiento computacional.
- Marcos de referencia.

Tabla 9. Finalidad de los artículos. Fuente: Elaboración propia.

Finalidad	Nº
Métodos o actividades de promoción.	11
Propuestas o análisis de la integración.	6
Análisis/Críticas/Definiciones.	26
Marcos de referencia.	2

Los resultados de la Tabla 9 son obtenidos del análisis realizado a los 45 artículos seleccionados. De los cuales el 60% corresponden a la literatura que realiza análisis, críticas o definiciones del propio concepto. Por otro lado el 25% de los artículos corresponden a métodos o actividades que promueven el pensamiento computacional. Relegando un 15% para la integración en el currículum y, en menor medida, para los marcos de referencia.

Continuando con los resultados obtenidos por Kalelİođlu, et. al (2016) podemos observar como únicamente ha sufrido un gran aumento de, prácticamente, un 50% los artículos cuya finalidad es realizar un análisis, crítica o definición, mientras que los métodos o actividades han obtenido un ligero aumento del 2%. Por otro lado, tanto los marcos de referencia como las propuestas de inclusión se han visto reducidas en más de un 50% su producción.

Datos que reflejan un claro aumento del interés hacia el pensamiento computacional, en estos tres últimos años. Mas dicho interés es mostrado fijando su foco en las partes más prácticas analizando, realizando críticas y definiciones del pensamiento computacional, así como con los métodos o actividades llevadas a cabo para promoverlo entre los discentes.

Artículos como el Art3, Art8 o Art30 muestran la gran selección de herramientas digitales de que disponen los docentes para trabajar el pensamiento computacional o los Art6, Art12 y Art45 que exploran las asignaturas STEM a través de la robótica o lenguajes de programación visual. Unos hechos que, además, de remarcar el carácter práctico de dichos artículos, muestran, al mismo tiempo, como dirigen el desarrollo del pensamiento computacional ,en su gran mayoría, hacia las asignaturas anteriormente mencionadas.

Mientras tanto, la parte más teórica queda relegada a un segundo plano. Posiblemente debido a la inclusión del término en los currículos oficiales por parte de un gran número de países, que ya han comenzado, incluso, a potenciarlo con programas nacionales. Así, entre los artículos seleccionados podemos encontrar casos concretos como son el Art2, Art4 y el Art12 que proponen currículos muy específicos, alejados de lo que proponen las administraciones. Una hipótesis que corrobora documentos como el Art 24, en cuyo contenido podemos encontrar cómo la Comunidad de Madrid incluyó el pensamiento computacional en su currículo académico de forma muy genérica. Unos hechos que podrían explicar la disminución sufrida por estas finalidades.

## **8.2. Análisis del rango académico al que van dirigidos los artículos.**

Los rangos académicos es España y, gran parte de Europa, los podemos dividir desde Educación Infantil hasta la Educación Superior, cuatro rangos que distribuyen a los discentes desde los 0-3 años hasta la edad en la que deseen dejar la educación formal, lo cual abarca un gran abanico de edades.

De los 45 artículos científicos seleccionados únicamente 2 quedan fuera del rango académico al estar dirigidos hacia un perfil concreto: los docentes. De los 43 documentos restantes, tras ser analizada la población a la que van dirigidos, 12 corresponderían a la Educación Superior, tan sólo 3 a la Educación Secundaria, 19 a la Educación Primaria y un único artículo referente a la Educación Infantil.

Tabla 10. Distribución de los rangos académicos. Fuente: Elaboración propia.

Rango académico	Nº
Educación Superior	12
Educación Secundaria	3
Educación Primaria	19
Educación Infantil	1
Docentes	2

Continuando el estudio realizado por Kalelloğlu (2016) se puede apreciar como aunque existe un leve aumento correlativo, se mantienen los niveles de investigación en Educación Primaria y Educación Superior como referentes de investigación.

Queremos que los niños crezcan sin la preocupación de cometer errores. Queremos que la gente juegue, pruebe cosas nuevas, y cuando las cosas fallan se adapten y lo corrijan. Esa es la forma, el espíritu de hacer las cosas (Resnick, 2015)

Obtenemos como resultado que desde 2006 las investigaciones se focalizan en estos dos grandes grupos principales. La lógica explica el primer resultado dado el gran interés por potenciar el pensamiento computacional desde edades tempranas, un hecho que tendría como consecuencia un mayor desarrollo del mismo y su posterior uso en el resto de rangos de edad.

Asimismo, incorporar el concepto en las últimas etapas de la educación formal tiene como consecuencia el desarrollo de un pensamiento computacional más especializado en determinadas asignaturas. Además, hay dos artículos cuyos participantes son docentes en activo como en el Art 40 o el Art17, un documento donde se presentan a profesores y profesoras las herramientas digitales más actuales.

Aunque no hay propuestas de integración del pensamiento computacional en el currículo de todos los futuros docentes. Sin embargo, sería interesante poder estudiar más artículos científicos cuyos participantes fueran estudiantes de educación como en Art16 donde sus autores proponen un curso que introduce a futuros docentes de educación primaria en los conceptos básicos: organización, aprendizaje, motivación y evaluación.

### 8.3. Metodologías o tipos de artículos encontrados.

Además, de poder comprobar cómo ha sido estructurada una investigación, qué han tenido en cuenta por los autores y qué han decidido obviar, la metodología utilizada en un artículo nos indica las posibles técnicas e instrumentos utilizados durante su investigación. Los resultados obtenidos tras realizar el análisis a los artículos seleccionados puede verse en la Tabla 11.

Tabla 11. Metodologías seguidas por los artículos. Fuente: Elaboración propia.

Metodología	Nº
Teóricos	14
Cuantitativos	5
Cualitativos	11
Revisiones	3
Mixtos	12

Al comparar los datos de la Tabla 11 con los resultados mostrados por el artículo de Kalelİođlu, et. al (2016) tiene un sorprendente fruto, debido a que los porcentajes son exactamente iguales. Por tanto, podemos afirmar que la tendencia de los documentos seleccionados se han mantenido desde 2006 hasta la actualidad.

El 31% de los documentos presentan una metodología de enfoque teórico donde los autores suelen analizar, criticar, reflexionar, realizan propuestas de posibles investigaciones o modificaciones del currículo. El 11% de los artículos han desarrollado una metodología cuantitativa, la cual se basa en el estudio estadístico de los datos obtenidos. Los cuales se consiguen a través de instrumentos dirigidos al resultado deseado, para realizar un análisis numérico.

Por otro lado, el 24% corresponde a la cualitativa. Una metodología que hace uso de grupos de discusión, entrevistas y observación donde para obtener los resultados se realiza en base a la interpretación de las informaciones. Las revisiones tiene un 6% de la tendencia, aún cuando es una metodología en esplendor como se comentó con anterioridad. El último 26% pertenece a los estudios que utilizan una técnica mixta, que hace uso de las metodologías cuantitativa y cualitativa.

Estas tendencias, además, servirán de cara a futuras posibles investigaciones, dado que nos permite determinar posibles espacios de investigación existentes. Los cuales, también podrían convertirse en artículos de verdadero impacto al partir, por ejemplo, de las opciones con menor tendencia, una variable más a tener en cuenta.

#### **8.4. Propuestas de introducción del pensamiento computacional en el currículum.**

De los 45 artículos seleccionados en esta revisión sistemática seis de ellos corresponden a propuestas de introducción del pensamiento computacional en el currículum oficial. Frente a los 39 artículos que no realizan crítica o propuesta al mismo.

Tabla 12. Artículos con o sin propuestas de introducción del pensamiento computacional en el currículum. Fuente: Elaboración propia.

Tipo	Nº
Con propuestas	6
Sin propuestas	39

Aunque hemos comentado con anterioridad las posibles razones de tal resultado es de obligado cumplimiento realzar la importancia de este tipo de artículos, pues gracias a ellos podemos saber si las administraciones públicas están implementando correctamente o no, si debemos implementarlo o no en diferentes rangos de edad o, sencillamente, cómo debemos implementarlo.

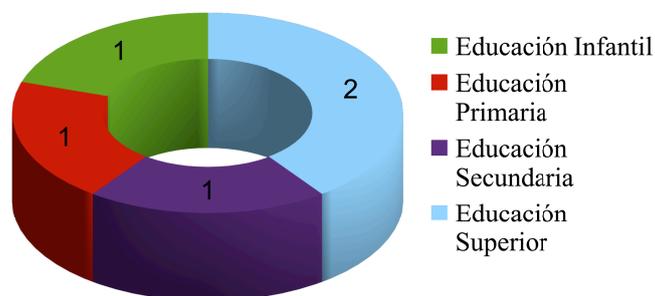


Gráfico 3. Propuestas curriculares distribuidas por rangos académicos. Fuente: Elaboración propia.

Los rangos de edad a los que van dirigidos las diferentes propuestas curriculares pueden ser contemplados en el Gráfico 3. Aunque existe una ligera ventaja para la Educación Superior, puede observarse con total claridad como esta revisión ha obtenido prácticamente un ejemplo de propuesta para cada uno de los rangos académicos.

Our purpose of the research was to propose what types of tasks could be made accessible and meaningful for assessing students' CT. To that end, we presented the three dimensions and six types of tasks that could assess different dimensions of CT (Zhong, Wang, Chen, & Li, 2015, p. 24).<sup>10</sup>

Cabe destacar los Art2, Art4 y Art 12 debido a sus propuestas de marco de referencia para desarrollar el pensamiento computacional a través del currículo formal. Aunque el Art2 lo realiza de forma teórica lo complementa con un apartado dedicado al conocimiento que los docentes necesitarían para poder enseñar ese mismo currículo. El Art4 y el Art12 si que proponen de una forma práctica el desarrollo específico de una serie de tareas o lecciones a través de lenguajes de programación visual, cada uno para el rango de edad específico.

<sup>10</sup> Nuestro propósito en la investigación fue proponer qué tipos de tareas podrían hacerse accesibles y significativas para valorar el pensamiento computacional de los estudiantes. A tal fin, se presentaron las tres dimensiones y seis tipos de tareas que podrían evaluar las diferentes dimensiones del pensamiento computacional (Zhong, Wang, Chen, & Li, 2015, p. 24)

[...] existe una descripción muy pormenorizada del currículo, una característica frecuente en la legislación educativa española, que deja un margen muy estrecho a docentes y centros educativos para contextualizar y adaptar los diseños curriculares. Especialmente, los criterios de evaluación poseen un nivel de concreción tan elevado y son tan numerosos, que condicionan la metodología docente y sus posibilidades de crear entornos de aprendizaje flexibles (Valverde, et al., 2015 p. 12).

El cuarto artículo el Art24 realiza una comparación con lo realizado en el extranjero y con lo realizado en la Comunidad Autónoma de Madrid, donde encontramos una propuesta curricular demasiado encorsetada y poco eficaz. Únicamente queda mencionar la curiosidad que presentan el Art11 y el Art32 que realizan una propuesta muy similar de inclusión del pensamiento computacional en la Biología, ambos lo hacen de forma teórica.

### **8.5. Experiencias reales en educación primaria.**

Tabla 13. Artículos con o sin experiencias reales. Fuente: Elaboración propia.

Tipo	Nº
Si	13
No	32

Este último tema de la investigación recaba las experiencias llevadas a cabo en las aulas de los diferentes rangos académicos. Tal y como recoge la Tabla 13, una vez realizado el análisis se aprecia como el 29% de los documentos recogen las experiencias reales llevadas a cabo con diferentes intereses de investigación. Mientras que un alto porcentaje, el 71% de los artículos científicos, no tienen una experiencia real. De nuevo un espacio amplio donde seguir investigando.

The Categorizer is a specific ongoing project, but reflects a broader goal shared by the educational technology community: to engage students in STEM knowledge construction practices (Wilkerson-Jerde, 2014, p. 116).<sup>11</sup>

<sup>11</sup> El Categorizador es un proyecto específico en curso, pero que refleja un objetivo compartido por la comunidad tecnológica educativa: involucrar a los estudiantes en prácticas de construcción del conocimiento de CTIM.

Cabe destacar que la mayoría de los artículos tienen como finalidad potenciar el pensamiento computacional con el objetivo de mejorar los resultados en el ya denominado STEM. Una relación lineal que llega a rozar con la rutina de lo escrito estos últimos años. Ejemplo de lo afirmado serían los Art18, Art28, Art33 o Art43 entre otros.

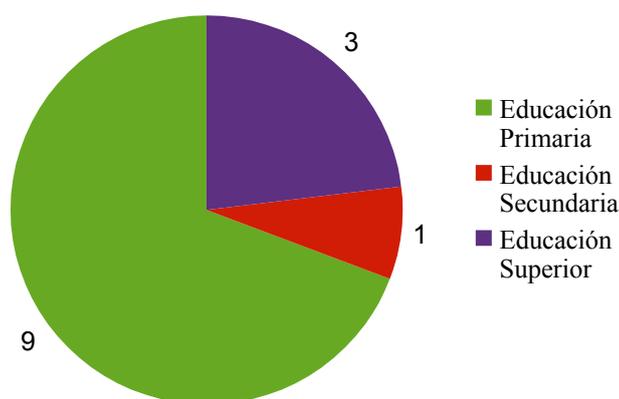


Gráfico 4. Experiencias reales distribuidas en rangos académicos. Fuente: Elaboración propia.

En cuanto al análisis en base a los rangos académicos se aprecia como el 70% de las experiencias se desarrollan en Educación Primaria, mientras que el 23% tienen lugar en la Educación Superior. Dejando el restante 7% a al desarrollado en la secundaria. Volviendo nuestra mirada a los temas anteriormente analizados podemos comprobar como se mantiene una coherencia entre ellos y el actual. Aunque debemos recordar que sólo el Art16 está enfocado a futuros docentes o estudiantes de educación.

Unos resultados que nos muestran una clara diferenciación entre los denominados “low-floor” o niveles iniciales y los “high-celiling” o niveles avanzados. Podemos apreciar como las investigaciones y prácticas van orientadas a la iniciación del pensamiento computacional y no tanto al desarrollo más especializado. Parece lógico comenzar por propuestas de implantación y así asentar unos cimientos sobre los que construir una especialización llegado el momento.

Son múltiples las herramientas digitales que se utilizan en estas experiencias, aunque debemos destacar el Art30. Los autores hacen uso de dos herramientas bien diferenciadas: Scratch y Second Life. La mezcla de la red social con el lenguaje de programación visual da lugar a un híbrido denominado: Scratch4SL. Esta unión permite a los usuarios codificar en tres dimensiones, una nueva experiencia a tener en cuenta.

Sin embargo, debemos hacer una especial mención al Art34, al menos, por dos razones. En primer lugar, realizan un análisis sobre la influencia del género en el desarrollo del pensamiento computacional. Los resultados obtenidos antes y después del proceso muestran puntuaciones más altas, en ambos casos, para las chicas frente a los chicos que obtienen menor puntuación pero sufren un mayor desarrollo. Un resultado que topa con una realidad donde los grados relacionados con CTIM son cursados en su gran mayoría por hombres.

En segundo lugar, el artículo científico tiene como herramienta principal Electron Toolset, un editor del juego Neverwinter Nights 2, un juego de rol, basado en la afamada factoría Dungeons and Dragons, publicado por Atari en 2006. Esta herramienta permite crear infinidad de mundos desde cero, donde el único límite es tu imaginación. Sin embargo, frente a otros estudios el Art34 tiene como hilo conductor la historia, un cuento o novela que los jugadores deberán recorrer.

Over the past several years, *computational literacy* has become an important topic for discussion for K12 STEM education. Developing computational literacy requires developing epistemic and representational practices such as thinking algorithmically, and designing and creating computational artifacts such as programs and simulations. Still, computational literacy does not yet have any noticeable representation in the standard scope and sequence of public schools, especially at the elementary level. (Farris & Sengupta, 2016, p. 3)<sup>12</sup>

---

12 En los últimos años, la alfabetización computacional se ha convertido en un tema importante de discusión para la educación primaria entorno a CTIM. El desarrollo de la alfabetización computacional requiere el desarrollo epistémico y las prácticas de representación tales como el pensamiento algorítmico, el diseño y la creación de artefactos computacionales, tales como programas y simulaciones. Aún así, la alfabetización computacional todavía no tiene ninguna representación notable en el ámbito y resultados estándar de las escuelas públicas, especialmente en el nivel elemental. (Farris & Sengupta, 2016, p. 3)

A título meramente anecdótico cabe destacar el caso real de Naomi Novik, una jugadora de Neverwinter que comenzó a mostrar curiosidad por la creación de historias y le llevó a estudiar un curso de postgrado de informática en la Universidad de Columbia. Curso que no llegó a terminar dado que comenzó a trabajar como diseñadora y desarrolladora del propio Neverwinter. Cuando finalizó su trabajo en la compañía forjó la serie de libros Temerario convirtiéndose en una afamada escritora de literatura juvenil. Reincidir en el carácter anecdótico, dado que no existe un estudio o investigación que pruebe la relación, de nuevo un campo en el que seguir investigando.

## **9. Conclusiones**

El propósito del presente estudio fue la realización de una revisión sistemática, de los últimos tres años, sobre el concepto del pensamiento computacional en la educación formal. Ante unos resultados en su mayor parte positivos, mejorables con una mayor profundidad en dos de sus temas: propuestas curriculares y experiencias reales, puede recomendarse el uso particular del presente documento como base para realizar futuras investigaciones. De este modo, supondría una mejora de la calidad en los procesos iniciales de dichos proyectos.

La presente propuesta ha mostrado la importancia de las revisiones sistemáticas. Una afirmación basada en la perspectiva de unos síntomas que indican la adaptación de una serie de cambios en los sistemas educativos, los cuales comenzaron su gestación varios años atrás y se ha visto incrementada a través del auge sufrido en esta última década, debido a la necesidad de educar pensando en el futuro. Una gestación que se ha visto potenciada por las nuevas herramientas que se han sumado a las ya existentes.

Por suerte nacimos para aprender, lentamente, de forma segura, tropiezas, resbalas, te arrastras, fallas y caes, hasta que un día caminas. Cada equivocación hace que tu cerebro sea un poco más fuerte. Fallar es sólo otra forma de decir crecer, pero continua. Eso es aprender. (Salman, 2015)

Además, en este análisis de los resultados obtenidos, el conjunto de artículos proponen o tienen una filosofía donde el error debe pasar a ser parte fundamental del proceso de aprendizaje, mientras el estudiante avanza con autonomía al ritmo que él mismo se establece. Asimismo, muestran al docente como un guía dentro del proceso de enseñanza-aprendizaje, siendo el discente quien planifica sus actividades y trabaja en sus centros de interés.

Entre los objetivos fijados en la presente investigación podemos observar como se ha llevado a cabo el *Estudio de las diferentes temáticas desde las que se estudia el pensamiento computacional*, a través del análisis de la compilación de documentos seleccionados. Obtuvimos las tendencias desarrolladas en los últimos tres años, unos resultados que expusimos en una tabla, subdividida en cuatro grupos temáticos que permitieron una mejor comprensión de los mismos.

En cuanto al *Examen de los diferentes niveles a los que van dirigidos los artículos* se realizó un proceso similar al anterior. Se extrajo la posibilidad de corresponder al año de publicación 2014, 2015 o 2016, debido a las revisiones sistemáticas realizadas en los últimos años. Para lograr el objetivo específicos dichos datos se expusieron en una tabla que recogía los diferentes rangos académicos a los que podía ir dirigido el documento elegido: Educación Infantil, Educación Primaria, Educación Secundaria, Educación Superior, e incluso se incluyó la denominación Docente para clasificar aquellas investigaciones que tenían como participantes tanto a futuros profesores o profesoras como aquellos y aquellas que están en activo.

El objetivo que pretendía *Analizar las metodologías seguida por los diferentes proyectos* se realizó a través de la lectura profunda de los distintos artículos científicos seleccionados. Como resultado de dicho análisis se mostraron los resultados en una tabla que recogía las cinco opciones posibles dentro de una metodología de investigación: un artículo teórico o ideario, cualitativo, cuantitativo, una revisión o un mixto. Teniendo en cuenta siempre que la revisión se ha realizado de documentos indexados estos resultados nos muestran posibles espacios de investigación.

La función de un maestro no ha de limitarse a transmitir sus conocimientos a los alumnos. No es ese el concepto de maestro que yo tengo. Un maestro debe, ante todo, aguijonear la curiosidad de los niños y niñas, ofrecerles la posibilidad de descubrir por sí mismos hechos o realidades que permanecen ocultos hasta que ellos abren esa puerta. La curiosidad es la herramienta más valiosa que tenemos los seres humanos para aprender, y no siempre se impulsa en la escuela. Demasiados datos, demasiadas áreas contra reloj, demasiada presión desde arriba y demasiada programación [...] La educación ha de ser un vehículo para impulsar capacidades y no un corsé que ahogue potenciales. (Bona, 2013, p.21)

Para llevar a cabo el *Examen de las propuestas de inclusión del concepto en el currículum por los diferentes artículos* se aplicó un criterio más restrictivo a los documentos seleccionados, para así únicamente realizar una reflexión de aquellos que realizaban propuestas concretas. En esta ocasión a la tabla que recogía el número de artículos que cumplían con el criterio se realizó una gráfica que mostraba la distribución de esas propuestas en base al rango académico. La reflexión de las propuestas mostraron como no se trata de introducir el concepto como un conocimiento a transmitir, sino como la creación de un espacio para la curiosidad, la creatividad y la iniciativa.

El último objetivo específico *Estudiar aquellas experiencias reales llevadas a cabo en espacios de educación formal* se llevó a cabo de una forma muy similar al anterior, pues tras especificar un criterio más concreto se realizó una revisión de los documentos seleccionados incluyendo únicamente aquellos que realizaban experiencias reales. Los resultados fueron expuestos en forma de tabla, que recogía el volumen de artículos que cumplían con el criterio. Además, se generó una gráfica para interpretar los rangos académicos en los que se desarrollaban dichas experiencias.

La consecución, por tanto, de los cinco objetivos específicos propició el hecho de cumplir el único objetivo general que consistía en Realizar una revisión sistemática del concepto pensamiento computacional a través del análisis de casos, estudios y experiencias realizadas. En el conjunto de la investigación se mantuvo un estricto seguimiento de la metodología, cumpliendo con los criterios de búsqueda, con las bases de datos seleccionadas y las razones de exclusión. Así se obtuvo como resultado la selección de cuarenta y cinco artículos científicos.

Gracias a esto y al reflexivo análisis realizado, se pudo registrar de forma visual una tabla que contenía el código del artículo, el título, el primer autor, la fecha de publicación, el idioma, la función, el rango académico, la metodología, si contenía propuestas o experiencias reales o no. Un catálogo que facilitó el desarrollo del resto de objetivos, al mismo tiempo que permitía la conclusión del mismo. Además se acompañó a la tabla con dos gráficas que mostraban de forma visual dos aspectos no incluidos entre los objetivos, como son la fecha de publicación y el idioma, pero que permitían complementar la investigación realizada.

Esta investigación ha cumplido con los requisitos, los objetivos y la metodología, para ser considerada una investigación objetiva que, por tanto, servirá como base para futuros estudios basados en pensamiento computacional. Dado que el estudio ha mejorado nuestro conocimiento del estado actual del término a través de los artículos científicos publicados en los últimos tres años.

El único límite, relacionado con las búsquedas que existe son los índices de las diferentes bases de datos utilizadas. Es un hecho que no todos los artículos publicados son indexados y por tanto centenares de documentos han quedado excluidos de la presente revisión. Los cuales provocarían cambios sustanciales en los resultados obtenidos. Un límite que no puede ser subsanado sin incumplir los requisitos de calidad, que estipulamos en artículos publicados en revistas científicas.

Además los resultados en este proyecto están sujetos al menos a dos limitaciones. Por un lado, aunque se ha especificado todo el proceso de búsqueda y valoración de la información, las revisiones sistemáticas son herramientas metodológicas que precisan de una práctica, de una experiencia y conocimientos en los métodos de búsqueda y revisión, la presente es la primera para su autor, así como en la aplicación e interpretación de los resultados obtenidos.

La segunda limitación tiene que ver con la profundidad de estudio realizado en los temas predispuestos para ello. Tanto en las propuestas curriculares como en las experiencias reales se podría haber indagado más. Especificando temas o términos más específicos hubiéramos obtenidos unos resultados más profundos sobre los avances en estos últimos tres años. Además, al ser propuestas o prácticas permitirían obtener un mayor número de datos que analizar.

Sin embargo, los hallazgos de este estudio mantienen, como hemos visto, una serie de implicaciones importantes para la práctica futura. Existe, por tanto, una clara necesidad de dar a conocer la presente revisión y, así, no sólo se pueda continuar o comenzar proyectos, sino que permitirá seguir realizando revisiones sistemáticas a partir de la presente dentro de al menos un par de años.

Finalmente, con la presente revisión sistemática se ha perseguido conocer el estado actual en el que se encuentra el pensamiento computacional en la educación formal, cuáles son los tipos de investigaciones y sus finalidades más recientes, los últimos hallazgos y experiencias. Sirva, pues, este Trabajo Fin de Máster como precedente o 元<sup>13</sup> de la tesis doctoral “Educación digital, redes sociales y nuevas ecologías del aprendizaje” que continuará trabajando el pensamiento computacional en la educación formal. Es por ello, que se incluirán los datos obtenidos en la presente investigación a la primera fase del mismo.

---

13 Kanji japones que recoge varios significados como: origen, comienzo, inicio, primeros pasos, etc.

## 10. Bibliografía

- Balagurusamy, E. (2010). *Fundamentals of Computers*. Tata Mcgraw Hill Education.
- Bona, C. (2013). *De cómo 12 niños y un maestro buscaron cambiar el mundo: el Cuarto Hocico* (1. ed). Castellón de la Plana: Ediciones Hades.
- Clark, J., Rogers, M. P., Spradling, C., & Pais, J. (2013). What, no canoes? Lessons learned while hosting a Scratch summer camp. *Journal of Computing Sciences in Colleges*, 28, 204e210.
- CSTA and ISTE (2011). Computational Thinking Leadership Toolkit, first edition 2011. Computer Science Teachers Association (CSTA) and International Society for Technology in Education (ISTE).
- Farris, A. V., & Sengupta, P. (2016). Democratizing Children's Computation: Learning Computational Science as Aesthetic Experience. *Educational Theory*, 66(1-2), 279-296. <http://doi.org/10.1111/edth.12168>
- Gardner, M.K., Feng, W.C. Broadening accessibility to computer science for K-12 education, in: Proceedings of the 15th annual conference on innovation and technology in computer science education, ACM, 2010, pp. 229-233.
- González, J. S. (2009). Del "E-Learning" Al "M-Learning": Una Academia En Cada "iPhone." *Telos: Cuadernos de Comunicación E Innovación*, 81, 122-128.
- Goode, J., Margolis, J., & Chapman, G. (2014). Curriculum is not enough: the educational theory and research foundation of the exploring computer science professional development model. En J. D. Dougherty & Association for Computing Machinery (Eds.), SIGCSE '14 Proceedings of the 45th ACM technical symposium on Computer science education (pp. 493-498). New York: A C M . R e c u p e r a d o a p a r t i r d e <http://dl.acm.org/citation.cfm?doid=2538862.2538948>
- Grover, S., & Pea, R. (2013). Computational thinking in K-12, a review of the state of the field. *Educational Researcher*, 42(1), 38e43.

- Hakkarainen, P. (2009). Designing and implementing a PBL course on educational digital video production: lessons learned from a design-based research. *Etr&d- Educational Technology Research and Development*, 57(2), 211-228. <http://doi.org/10.1007/s11423-007-9039-4>
- Han Koh, K., Basawapatna, A., Bennett V., & Repenning, A. (2010). Towards the automatic recognition of computational thinking for adaptive visual language learning. In *Proceedings of the 2010 Conference on Visual Languages and Human Centric Computing (VL/HCC 2010)* (pp. 59-66). Madrid, Spain: IEEE Computer.
- Johnson, L., Adams Becker, S., Estrada, V., & Freeman, A. (2014). *NMC Horizon Report: 2014 K-12 edition*. Austin, Texas: The New Media Consortium. Retrieved from <http://www.nmc.org/pdf/2014-nmc-horizon-report-he-EN.pdf>
- Kafai, Y. B., & Burke, Q. (2014). *Connected code: why children need to learn programming*. MIT Press.
- Kalelıoğlu, F., Gülbahar, Y., & Kukul, V. (2016). A Framework for Computational Thinking Based on a Systematic Research Review. *Baltic J. Modern Computing*, 4(3), 583-596.
- Khan, Salman. (2015). *Salman Khan, el responsable de que millones de niños mejoren sus notas del colegio*. Recuperado a partir de [https://www.youtube.com/watch?time\\_continue=1&v=6\\_vN3BZnlnc](https://www.youtube.com/watch?time_continue=1&v=6_vN3BZnlnc)
- Lambert, L., Guiffre, H. Computer science outreach in an elementary school, *Journal of Computing Sciences in Colleges*, 24 (2009) 118-124.
- Lye, S. 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. <http://doi.org/10.1016/j.chb.2014.09.012>
- Manterola, C., Astudillo, P., Arias, E., & Claros, N. (2013). Revisiones sistemáticas de la literatura. Qué se debe saber acerca de ellas. *Cirugía Española*, 91(3), 149-155. <http://doi.org/10.1016/j.ciresp.2011.07.009>

- Olabe, X. B., Basogain, M. Á. O., & Basogain, J. C. O. (2015). Pensamiento Computacional a través de la Programación: Paradigma de Aprendizaje. *Revista de Educación a Distancia*, 0(46).
- Pea, R. D., & Kurland, D. M. (1984). On the Cognitive Effects of Learning Computer Programming: A Critical Look. Technical Report No. 9.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Siver, J., Silverman, B., y Kafay, Y. (2009). *Scratch: Programming for all*. Communications of the ACM, 52 (1), 60-67.
- Resnick, Mitch. (2015). *Scratch, el software gratuito del MIT con el que tu hijo aprenderá a programar*. Recuperado a partir de [https://www.youtube.com/watch?time\\_continue=110&v=moxOrgbrWw4](https://www.youtube.com/watch?time_continue=110&v=moxOrgbrWw4)
- Revans, R. (1979). Action Learning It's Terms and Character. *Management Decision*, 21, 1-6.
- Sánchez Meca, J. (2010). Cómo realizar una revisión sistemática y un meta-análisis. *Aula abierta*, 38(2), 53-64.
- ScratchEd Team [Portal Web] (2015). Computational Thinking webinars. Recuperado 2 de Junio de 2015, de <http://scratched.gse.harvard.edu/content/1488>
- Sengupta, P., Kinnebrew, J. S., Basu, S., Biswas, G., & Clark, D. (2013). Integrating computational thinking with K-12 science education using agent-based computation: A theoretical framework. *Education and Information Technologies*, 18(2), 351–380.
- Stošić, L. & Bogdanović, M. (2013). M-learning – A new form of learning and education. *International Journal of Cognitive Research in Science, Engineering and Education (IJCRSEE)*, 1(2), 114–118.
- Valverde Berrocoso, J., Fernández Sánchez, M.R., Garrido Arroyo, M.C. (2015). El pensamiento computacional y las nuevas ecologías del aprendizaje. *RED, Revista de Educación a Distancia*. 46(3). Septiembre de 2015.

- Wilkerson-Jerde, M. H. (2014). Construction, categorization, and consensus: student generated computational artifacts as a context for disciplinary reflection. *Etr&d- Educational Technology Research and Development*, 62(1), 99-121. <http://doi.org/10.1007/s11423-013-9327-0>
- Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-36.
- Wing, J. (2011). Research notebook: Computational thinking – What and why? *The Link Magazine*, Spring. Carnegie Mellon University, Pittsburgh.
- Záhorec, J., Hašková, A., & Munk, M. (2014). Assessment of Selected Aspects of Teaching Programming in SK and CZ. *Informatics in Education*, 13(1), 157-178.
- Zhong, B., Wang, Q., Chen, J., & Li, Y. (2015). An Exploration of Three-Dimensional Integrated Assessment for Computational Thinking. *Journal of Educational Computing Research*, 735633115608444. <http://doi.org/10.1177/0735633115608444>

### **10.1. Referencias documentales objeto de la revisión**

- Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge. *Educational Technology & Society*, 19(3), 47-57.
- Arellano, N., Rosas, M. V., Zuñiga, M. E., Fernández, J., & Guerrero, R. (2014). Una experiencia en la enseñanza de la programación para la permanencia de los alumnos de Ingeniería Electrónica. *Enseñanza y aprendizaje de ingeniería de computadores: Revista de Experiencias Docentes en Ingeniería de Computadores*, (4), 69-80.
- Basawapatna, A. (2016). Alexander Meets Michotte: A Simulation Tool Based on Pattern Programming and Phenomenology. *Educational Technology & Society*, 19(1), 277-291.
- Berland, M., & Wilensky, U. (2015). Comparing Virtual and Physical Robotics Environments for Supporting Complex Systems and Computational Thinking. *Journal of Science Education and Technology*, 24(5), 628-647.

- Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72, 145-157. <http://doi.org/10.1016/j.compedu.2013.10.02>
- Bordignon, F., & Iglesias, A. A. (2016). Más allá de las pantallas: experiencias en diseño y programación de objetos interactivos digitales. *Virtualidad, Educación y Ciencia*, 7(12), 49-58.
- Buckingham, L., & Hogan, J. M. (2014). Computational Science for Undergraduate Biologists via QUT.Bio.Excel. *Procedia Computer Science*, 29, 1403-1412. <http://doi.org/10.1016/j.procs.2014.05.127>
- Chao, P.-Y. (2016). Exploring students' computational practice, design and performance of problem-solving through a visual programming environment. *Computers & Education*, 95, 202-215. <http://doi.org/10.1016/j.compedu.2016.01.010>
- Cotik, V., & Monteverde, H. (2016). Evolución de la enseñanza de la informática y las TIC en la Escuela Media en Argentina en los últimos 35 años. *Virtualidad, Educación y Ciencia*, 7(12), 11-33.
- Csernoch, M., Biro, P., Math, J., & Abari, K. (2015). Testing Algorithmic Skills in Traditional and Non-Traditional Programming Environments. *Informatics in Education*, 14(2), 175-197.
- Czerkawski, B. C., & Lyman, E. W. (2015). Exploring Issues about Computational Thinking in Higher Education. *TechTrends: Linking Research and Practice to Improve Learning*, 59(2), 57-65. <http://doi.org/10.1007/s11528-015-0840-3>
- DeSchryver, M. D., & Yadav, A. (2015). Creative and Computational Thinking in the Context of New Literacies: Working with Teachers to Scaffold Complex Technology-Mediated Approaches to Teaching and Learning. *Journal of Technology and Teacher Education*, 23(3), 411-431.

- Farris, A. V., & Sengupta, P. (2016). Democratizing Children's Computation: Learning Computational Science as Aesthetic Experience. *Educational Theory*, 66(1-2), 279-296. <http://doi.org/10.1111/edth.12168>
- Grout, V., & Houlden, N. (2014). Taking Computer Science and Programming into Schools: The Glyndŵr/BCS Turing Project. *Procedia - Social and Behavioral Sciences*, 141, 680-685. <http://doi.org/10.1016/j.sbspro.2014.05.119>
- Howland, K., & Good, J. (2015). Learning to communicate computationally with Flip: A bi-modal programming language for game creation. *Computers & Education*, 80, 224-240. <http://doi.org/10.1016/j.compedu.2014.08.014>
- Imberman, S. P., Sturm, D., & Azhar, M. Q. (2014). Computational Thinking: Expanding the Toolkit. *J. Comput. Sci. Coll.*, 29(6), 39-46.
- Israel, M., Pearson, J. N., Tapia, T., Wherfel, Q. M., & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. *Computers & Education*, 82, 263-279. <http://doi.org/10.1016/j.compedu.2014.11.022>
- Israel, M., Wherfel, Q. M., Pearson, J., Shehab, S., & Tapia, T. (2015). Empowering K-12 Students with Disabilities to Learn Computational Thinking and Computer Programming. *TEACHING Exceptional Children*, 48(1), 45-53. <http://doi.org/10.1177/0040059915594790>
- Jenson, J., & Droumeva, M. (2016). Exploring Media Literacy and Computational Thinking: A Game Maker Curriculum Study. *Electronic Journal of E-Learning*, 14(2), 111-121.
- Jon Good. (2016). Computational Thinking as a Computer Science Education Framework and the Related Effects on Gender Equity, 277-278.
- Jun, S., Han, S., Kim, H., & Lee, W. (2014). Assessing the computational literacy of elementary students on a national level in Korea. *Educational Assessment Evaluation and Accountability*, 26(4), 319-332. <http://doi.org/10.1007/s11092-013-9185-7>

- Kafai, Y. B., & Burke, Q. (2015). Constructionist Gaming: Understanding the Benefits of Making Games for Learning. *Educational Psychologist*, 50(4), 313-334. <http://doi.org/10.1080/00461520.2015.1124022>
- Kara, L. (2015). A Critical Look at the Digital Technologies in Architectural Education: When, where, and how? *Procedia - Social and Behavioral Sciences*, 176, 526-530. <http://doi.org/10.1016/j.sbspro.2015.01.506>
- Lee, A. (2015). Determining the effects of computer science education at the secondary level on STEM major choices in postsecondary institutions in the United States. *Computers & Education*, 88, 241-255. <http://doi.org/10.1016/j.compedu.2015.04.019>
- Lewis, C., Esper, S., Bhattacharyya, V., Fa-Kaji, N., Dominguez, N., & Schlesinger, A. (2014). Children's Perceptions of What Counts As a Programming Language. *J. Comput. Sci. Coll.*, 29(4), 123-133.
- Michaelson, G. (2015). Teaching Programming with Computational and Informational Thinking. *Journal of Pedagogic Development*, 5(1), 51-65.
- Miguel Zapata-Ros. (2015). Pensamiento computacional y alfabetización digital. *RED: Revista de Educación a Distancia*, (46). Recuperado a partir de <https://dialnet.unirioja.es/servlet/articulo?codigo=5284113>
- Mladenovic, S., Krpan, D., & Mladenovic, M. (2016). Using Games to Help Novices Embrace Programming: From Elementary to Higher Education. *International Journal of Engineering Education*, 32(1), 521-531.
- Park, S. Y., & Jeon, Y. (2015). Teachers' perception on Computational Thinking in Science Practices. *International Journal of Education and Information Technologies*, 9, 180-185.
- Pellas, N., & Peroutseas, E. (2016). Gaming in Second Life via Scratch4SL: Engaging High School Students in Programming Courses. *Journal of Educational Computing Research*, 54(1), 108-143. <http://doi.org/10.1177/0735633115612785>

- Promraksa, S., Sangaroon, K., & Inprasitha, M. (2014). Characteristics of Computational Thinking about the Estimation of the Students in Mathematics Classroom Applying Lesson Study and Open Approach. *Journal of Education and Learning*, 3(3), 56-66.
- Repenning, A., Webb, D. C., Koh, K. H., Nickerson, H., Miller, S. B., Brand, C., ... Repenning, N. (2015). Scalable Game Design: A Strategy to Bring Systemic Computer Science Education to Schools through Game Design and Simulation Creation. *Acm Transactions on Computing Education*, 15(2), 11. <http://doi.org/10.1145/2700517>
- Rodríguez, A. C. (2015). Incorporación de la programación informática en el currículum de biología. *Magister: Revista miscelánea de investigación*, 27(2), 76-82.
- Saez Lopez, J.-M., Roman-Gonzalez, M., & Vazquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using «Scratch» in five schools. *Computers & Education*, 97, 129-141. <http://doi.org/10.1016/j.compedu.2016.03.003>
- Sanford, J. F., & Naidu, J. T. (2016). Computational Thinking Concepts for Grade School. *Contemporary Issues in Education Research*, 9(1), 23-32.
- Snodgrass, M. R., Israel, M., & Reese, G. C. (2016). Instructional supports for students with disabilities in K-5 computing: Findings from a cross-case analysis. *Computers & Education*, 100, 1 - 17. <http://doi.org/10.1016/j.compedu.2016.04.011>
- Swaid, S. I. (2015). Bringing Computational Thinking to STEM Education. *Procedia Manufacturing*, 3, 3657-3662. <http://doi.org/10.1016/j.promfg.2015.07.761>
- Tahy, Z. S. (2016). How to Teach Programming Indirectly--Using Spreadsheet Application. *Acta Didactica Napocensia*, 9(1), 15-22.

- Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015). Computational Thinking in Compulsory Education: Towards an Agenda for Research and Practice. *Education and Information Technologies*, 20(4), 715-728. <http://doi.org/10.1007/s10639-015-9412-6>
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, 25(1), 127-147. <http://doi.org/10.1007/s10956-015-9581-5>
- Werner, L., Denner, J., & Campe, S. (2015). Children Programming Games: A Strategy for Measuring Computational Learning. *Acm Transactions on Computing Education*, 14(4), 24. <http://doi.org/10.1145/2677091>
- Wilkerson-Jerde, M. H. (2014). Construction, categorization, and consensus: student generated computational artifacts as a context for disciplinary reflection. *Etr&d- Educational Technology Research and Development*, 62(1), 99-121. <http://doi.org/10.1007/s11423-013-9327-0>
- Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). Computational Thinking in Elementary and Secondary Teacher Education. *ACM Transactions on Computing Education*, 14(1). <http://doi.org/10.1145/2576872>
- Zhong, B., Wang, Q., Chen, J., & Li, Y. (2015). An Exploration of Three-Dimensional Integrated Assessment for Computational Thinking. *Journal of Educational Computing Research*, 735633115608444. <http://doi.org/10.1177/0735633115608444>