

Fundamentos y perspectivas del pensamiento computacional: un análisis integral para la investigación futura

Foundations and perspectives of computational thinking: a comprehensive analysis for future research

Jorge Monge-Fallas¹, Liliana Sancho-Chavarría², César Garita-Rodríguez³, Antonio González-Torres⁴, Ignacio Trejos-Zelaya⁵

Fecha de recepción: 19 de febrero, 2024
Fecha de aprobación: 23 de mayo, 2024

Monge-Fallas, J; Sancho-Chavarría, L; Garita-Rodríguez, C; González-Torres, A; Trejos-Zelaya, I. Fundamentos y perspectivas del pensamiento computacional: un análisis integral para la investigación futura. *Tecnología en Marcha*. Vol. 38, N° 1. Enero-Marzo, 2025. Pág. 145-156.

 <https://doi.org/10.18845/tm.v38i1.7055>

- 1 Instituto Tecnológico de Costa Rica. Costa Rica.
 jomonge@itcr.ac.cr
 <https://orcid.org/0000-0002-1651-3543>
- 2 Instituto Tecnológico de Costa Rica. Costa Rica.
 lsancho@tec.ac.cr
 <https://orcid.org/0000-0002-4522-1259>
- 3 Instituto Tecnológico de Costa Rica. Costa Rica.
 cesar@itcr.ac.cr
 <https://orcid.org/0000-0003-4592-3266>
- 4 Instituto Tecnológico de Costa Rica. Costa Rica.
 antonio.gonzalez@tec.ac.cr
 <https://orcid.org/0000-0001-5427-0637>
- 5 Instituto Tecnológico de Costa Rica. Costa Rica.
 itrejos@tec.ac.cr
 <https://orcid.org/0000-0003-4361-8444>

Palabras clave

Pensamiento computacional; resolución de problemas; modelos algorítmicos; automatización de procesos; análisis de sistemas; evaluación educativa.

Resumen

Este artículo profundiza en la comprensión y aplicación del pensamiento computacional, una habilidad crucial en la era digital. Concebido inicialmente por Wing en 2006, este concepto no solo implica la codificación y programación de computadoras, sino que también se extiende a la resolución de problemas, el diseño de sistemas y la comprensión del comportamiento humano basado en los principios de la informática. Con el paso del tiempo, este enfoque ha evolucionado para incluir habilidades críticas como la abstracción, el análisis de datos, la modelización y la automatización de soluciones. Este estudio explora diversas interpretaciones del pensamiento computacional, desde ser una técnica para idear soluciones hasta una competencia intelectual para el análisis y solución de problemas reales. A través del análisis de literatura y evaluación de distintas herramientas y metodologías, la investigación destaca en particular la importancia de habilidades como la descomposición de problemas, abstracción, pensamiento algorítmico y evaluación en la educación y práctica profesional. A pesar de la ausencia de una definición unificada, el pensamiento computacional se reconoce como una habilidad multidimensional esencial, subrayando la necesidad de estrategias de enseñanza y evaluación efectivas. Además, se enfatiza la necesidad de establecer una definición universal de pensamiento computacional, seguida de la identificación clara de sus dimensiones y elementos constitutivos. El desarrollo de índices precisos y métricas objetivas es crucial para evaluar eficazmente el nivel de pensamiento computacional, lo que sentará las bases para una investigación rigurosa y metódica en este campo.

Keywords

Computational thinking; problem solving; algorithmic models; process automation systems analysis; educational evaluation.

Abstract

This article delves into the understanding and application of computational thinking, a pivotal skill in the digital era. Initially conceptualized by Wing in 2006, this concept transcends mere coding and programming to encompass problem-solving, system design, and understanding human behavior based on computer science principles. Over the years, this approach has broadened to include critical skills such as abstraction, data analysis, modeling, and solution automation. The study explores various interpretations of computational thinking, from being a technique for devising solutions to an intellectual competency for analyzing and solving real-world problems. Through literature analysis and evaluation of different tools and methodologies, the research emphasizes the importance of skills such as problem decomposition, abstraction, algorithmic thinking, and evaluation in education and professional practice. Despite the absence of a unified definition, computational thinking is recognized as an essential multidimensional skill, underscoring the necessity of effective teaching and assessment strategies. Moreover, it stresses the need to establish a universal definition of computational thinking, followed by a clear identification of its dimensions and constituent elements. Developing accurate indices and objective metrics is crucial for effectively assessing the level of computational thinking, laying the groundwork for rigorous and methodical research in the field.

Introducción

El pensamiento computacional se ha consolidado como un paradigma esencial en el campo de la educación y la investigación, ofreciendo una nueva perspectiva para abordar y resolver problemas complejos de una manera sistemática y eficiente. Esta habilidad multidimensional, que va más allá de la mera codificación y programación, involucra un conjunto de procesos mentales que permiten conceptualizar problemas y soluciones operables por máquinas, pero aplicables a un espectro mucho más amplio de disciplinas y contextos.

La noción de pensamiento computacional fue introducida y popularizada por Jeannette Wing en 2006 [1], marcando un hito en la conceptualización de cómo los individuos pueden integrar y aplicar los principios de la informática en la resolución de problemas, la construcción de sistemas y la interpretación del comportamiento humano. Esta concepción inicial ha sido el punto de partida para una exploración más profunda y detallada del tema, involucrando a académicos, educadores e investigadores en una búsqueda por definir, refinar y expandir nuestra comprensión del pensamiento computacional.

La definición de Wing [2], que inicialmente se centró en los aspectos prácticos de la resolución de problemas y la concepción de sistemas a través de la lente de la informática, ha evolucionado para abarcar un conjunto más amplio de habilidades cognitivas y procesos mentales. Organizaciones como la Computer Science Teachers Association (CSTA) y la International Society for Technology in Education (ISTE) han contribuido significativamente a esta evolución, ampliando el alcance del pensamiento computacional para incluir habilidades como la abstracción, el análisis de datos, la modelización, la automatización de soluciones y la generalización de métodos a diferentes tipos de problemas [3].

La relevancia del pensamiento computacional trasciende el ámbito de la informática, alcanzando diferentes campos y disciplinas. Se ha convertido en una habilidad crucial en la era digital, no solo para quienes se dedican a las ciencias de la computación, sino también para todo aquel que busca soluciones innovadoras y efectivas a problemas complejos. La capacidad de descomponer problemas, abstraer componentes esenciales, diseñar algoritmos y evaluar soluciones, constituye la esencia de esta competencia intelectual, permitiendo a los individuos no solo comprender y operar dentro del mundo digital, sino también transformarlo.

En la siguiente sección, se explorarán las diversas conceptualizaciones del pensamiento computacional, destacando las definiciones proporcionadas por figuras y entidades prominentes en el campo. Se analizarán los elementos comunes y distintivos de estas definiciones, las habilidades y competencias asociadas al pensamiento computacional, y cómo estas habilidades se reflejan y pueden ser evaluadas en entornos educativos y prácticos. Este análisis no solo aclara la naturaleza multifacética del pensamiento computacional, sino que también subraya su relevancia y aplicabilidad en una amplia gama de contextos, preparando el terreno para una comprensión más profunda y una implementación más efectiva de esta competencia clave en el siglo XXI.

Métodos

La metodología de la revisión sistemática de la literatura se estructuró en diversas etapas para garantizar una recopilación exhaustiva y pertinente de la información. La búsqueda de literatura se realizó en bases de datos académicas de renombre como ACM, ScienceDirect, Eric, IEEE Xplore, Directory of Open Journals (DOAJ) y JSTOR. Además, se revisaron las memorias de conferencias destacadas, incluyendo las del 18th Koli Calling International Conference on Computing Education Research y la 2018 ACM Conference on International Computing Education Research, entre otras fuentes relevantes.

Las palabras clave empleadas para la búsqueda fueron “computational thinking” y “computational thinking assessment”. El período de tiempo definido para la recolección de documentos abarcó desde 2006 hasta 2022, lo cual permitió obtener un corpus inicial de 80 documentos incluyendo artículos y libros. En el siguiente Cuadro 1 se muestra los resultados obtenidos

Cuadro 1. Resumen de Artículos por Temática y Fuente.

Theme	ACM	SienceDirect	ERIC	IEEE Xplore	DOAJ	JSTOR	Conferences	Otras	TOTAL
Computational Thinking	5	2	4	3	0	0	1	5	20
Assessment of Computational Thinking	9	1	2	2	1	0	1	11	27
Systematic Literature Reviews	0	2	3	0	0	0	0	4	9
Frameworks and Models	1	0	0	0	0	1	1	1	4
Teaching and learning Computational Thinking	4	5	3	3	0	0	0	2	17
Curriculum	1	0	0	0	0	1	0	1	3
TOTAL	20	10	12	8	1	2	3	24	80

Posteriormente, se procedió a una revisión minuciosa de los resúmenes de cada uno de los documentos para realizar una selección inicial. Este proceso de filtrado se enfocó en descartar aquellos trabajos que no alineaban con los objetivos específicos del proyecto. Los documentos seleccionados se clasificaron en varias temáticas: pensamiento computacional, enseñanza y aprendizaje del pensamiento computacional, marcos teóricos y conceptuales, evaluación del pensamiento computacional y revisiones sistemáticas de literatura. Dentro de este esquema, se tomó la decisión de no incluir aquellos documentos relacionados con currículos y los que se centraban específicamente en la enseñanza y aprendizaje del pensamiento computacional, manteniendo el enfoque del estudio en la conceptualización y evaluación del pensamiento computacional.

Este enfoque metódico y detallado en la revisión de la literatura aseguró que la investigación se fundamentara en una base de datos comprensiva y actualizada que nos permitió extraer información relevante que se detalla en las siguientes dos secciones.

Resultados

En esta sección se incluye un análisis de los principales conceptos y enfoques presentados por los documentos seleccionados en la sección anterior, clasificados en términos de artículos orientados a: conceptualización, revisión de literatura y evaluación del pensamiento computacional.

Conceptualización del pensamiento computacional

Como se mencionó anteriormente, el concepto de pensamiento computacional, que trasciende los límites tradicionales de la informática, fue inicialmente introducido por Wing en el año 2006 [80]. Wing conceptualizó el pensamiento computacional como un método comprensivo

que abarca la resolución de problemas, la concepción de sistemas y la interpretación del comportamiento humano, todo ello fundamentado en los principios esenciales de la informática. Más adelante, Wing [2] refinó esta concepción, describiéndolo como la serie de procesos cognitivos implicados en la articulación de problemas y en la estructuración de sus soluciones, de manera que puedan ser ejecutadas de forma óptima por un agente procesador de información. En 2011, la Computer Science Teachers Association (CSTA) y la International Society for Technology in Education (ISTE) [3], ampliaron esta visión, definiendo el pensamiento computacional como una metodología de resolución de problemas que comprende, pero no se limita a, la formulación de problemas de tal manera que puedan ser abordados mediante computadoras y otras herramientas tecnológicas; la organización y análisis sistemático de datos; la modelización de datos a través de abstracciones como modelos y simulaciones; la automatización de soluciones mediante el pensamiento algorítmico, que implica secuencias de pasos definidos y ordenados; la identificación, análisis e implementación de soluciones con el fin de alcanzar una combinación óptima de pasos y recursos; y la generalización y transferencia de esta metodología de resolución a una amplia gama de problemas. De manera específica, la definición operativa propuesta por ISTE y CSTA se destaca por seis características distintivas: abstracción, compilación y análisis de datos, representación de datos, solución automatizada, combinación eficaz de pasos y recursos, y capacidad de transferencia de soluciones.

Por otro lado, Romañ-González [4] concibe el pensamiento computacional como la habilidad de plantear y resolver problemas apoyándose en los fundamentos de la computación, empleando la lógica y sintaxis inherentes a los lenguajes de programación, como son las secuencias básicas, bucles, iteraciones, condicionales, funciones y variables. Asimismo, Dimos et al. [5] clasifican las distintas interpretaciones del pensamiento computacional en tres categorías alternativas: la primera lo considera como una técnica para idear soluciones a problemas que puedan ser representadas, procesadas y ejecutadas por cualquier agente, como una computadora; la segunda lo aborda como un método exclusivo de resolución de problemas; y la tercera lo percibe como una competencia intelectual aplicable al análisis y solución de problemas reales, mediante la adopción y aplicación de principios y métodos algorítmicos. En particular, identifican las competencias fundamentales del pensamiento computacional en términos de abstracción (el acto de hacer un artefacto más comprensible al minimizar los detalles innecesarios), algoritmos (un enfoque para alcanzar una solución mediante una definición clara de los pasos a seguir), descomposición (un modo de concebir los artefactos en términos de sus componentes individuales), generalización (relacionada con el reconocimiento de patrones, similitudes y conexiones, y la explotación de estas características) y evaluación (un método que valida la eficacia de una solución a un problema más que su corrección).

Varios autores han delineado el pensamiento computacional en términos de habilidades y competencias. Grover y Pea [6], por ejemplo, destacan la abstracción y generalización de patrones (incluyendo modelos y simulaciones), el pensamiento sistémico de la información, sistemas simbólicos y representaciones, nociones algorítmicas (como los diagramas de flujo de control), la descomposición estructurada de problemas (modularización), el pensamiento iterativo, recursivo y paralelo, la lógica condicional, la eficiencia y limitaciones de rendimiento, y la depuración y detección sistemática de errores. De forma similar, Bocconi et al. [7] enfatizan la abstracción, el pensamiento algorítmico, la automatización, la descomposición, la depuración y la generalización como elementos esenciales incluidos en el pensamiento computacional. Cabe destacar la presencia de elementos comunes en diversos estudios, como el de Seiter y Brendan [8], quienes señalan procedimientos y algoritmos, descomposición de problemas, paralelismo, sincronización, abstracción y la representación de datos como los aspectos más generales del pensamiento computacional. Estos conceptos son de naturaleza cognitiva, en tanto se

basan en características latentes de la actividad cognitiva y se centran en abstracciones del pensamiento computacional, cuya evidencia se puede encontrar en programas, a diferencia de las habilidades de procesamiento necesarias para crear y modificar dichos programas.

En 2012, dentro del contexto del lenguaje de programación Scratch, Brennan y Resnick [9] describieron las ideas principales que se aprenden al desarrollar el pensamiento computacional, que incluyen secuencias, paralelismo, bucles, condicionales, eventos, datos y operadores. Si bien estos son elementos básicos, estos autores también consideran aspectos más generales como las prácticas computacionales y las perspectivas computacionales. Las primeras se refieren al desarrollo de procedimientos que aplican conceptos computacionales para resolver problemas, mientras que las segundas sostienen que la naturaleza de resolver problemas computacionales permite a las personas evolucionar y percibir el mundo de manera diferente en cuanto a tecnologías digitales.

Chai et al. [10] definen el pensamiento computacional como un método que asiste a las personas en la resolución de problemas, diseño de sistemas y comprensión del comportamiento humano a través del uso de conceptos fundamentales de la computación. Entre las habilidades medidas se incluyen la abstracción y descomposición de problemas, paralelismo, pensamiento lógico, sincronización, control de flujo, interacción con el usuario, representación de datos y organización de código.

Aho [11] propuso el pensamiento computacional como una metodología de investigación para generar nuevos modelos informáticos subyacentes adecuados para formular problemas. Este enfoque divide el pensamiento computacional en seis habilidades distintas: descomposición (dividir el problema en partes más pequeñas), reconocimiento de patrones (identificar similitudes dentro o fuera de la situación del problema), abstracción (simplificar un problema complejo enfocándose solo en los detalles relevantes), algoritmo (definir los pasos necesarios en orden para resolver el problema), codificación (desarrollar un programa en un lenguaje comprensible para la computadora basado en el algoritmo) y depuración (corregir errores o mejorar el programa).

Revisión de literatura

El Ministerio de Educación de China emitió un currículum estándar en tecnologías de información para la educación secundaria y definió el pensamiento computacional como una actividad que utiliza el conocimiento para resolver problemas en el campo de las ciencias de la computación [12]. A partir del análisis de más de 580 documentos, se realizó un análisis de agrupación (clustering) de términos utilizados en la literatura sobre pensamiento computacional, proponiendo una estructura jerárquica de indicadores dividida en tres niveles, que a su vez se subdividen en categorías y subcategorías. En total, se identificaron más de 30 indicadores en el nivel más bajo, siendo los más relevantes del primer nivel (práctica de pensamiento computacional): abstracción, descomposición, identificación, transformación, codificación, diseño, modelado, combinación, procesamiento de datos, algoritmo, programación, prueba, evaluación, expansión y transferencia.

Otro estudio [13] sintetizó las definiciones de pensamiento computacional refiriéndose a los procesos de pensamiento que emplean los estudiantes para construir soluciones ejecutables por una computadora, aplicando varios conceptos y prácticas computacionales en la resolución de problemas. Los principales indicadores considerados incluyeron secuencia, condicionales, repetición y descomposición.

Un análisis de casi 40 artículos con el objetivo de unificar conceptos relacionados con el pensamiento computacional [14] sugiere que los estudios previos han mostrado poca concordancia entre investigadores, sectores gubernamentales y no gubernamentales sobre

una definición unificada de pensamiento computacional. Basándose en la revisión bibliográfica, definieron el pensamiento computacional como el proceso de pensamiento empleado para resolver problemas, que abarca elementos de descomposición del problema, abstracción y pensamiento algorítmico. Además, argumentan que a la definición de pensamiento computacional le falta el elemento de dependencia, definido como el conocimiento de la comprensión de las interrelaciones entre las diferentes secciones de un problema descompuesto.

Evaluación del pensamiento computacional

Romañ-González [4] considera el pensamiento computacional como un elemento esencial en todas las disciplinas STEM, destacando la importancia de comprender los elementos implicados en el pensamiento computacional, las habilidades que se desarrollan, las estrategias de enseñanza-aprendizaje que se pueden emplear para fomentar el pensamiento computacional, y una pregunta de interés para los investigadores: ¿cómo evaluar el pensamiento computacional?

A pesar de la evidente falta de consenso sobre la definición precisa de pensamiento computacional, sus habilidades o indicadores, es indiscutible que el pensamiento computacional se considera una competencia importante del siglo XXI, al nivel de competencias como la aritmética. Esta competencia, considerada de alto nivel y complejidad, está relacionada con el desarrollo de habilidades de pensamiento abstracto-matemático y con el enfoque pragmático-ingenieril. Por esta razón, en los últimos años, las tendencias en educación están reconociendo a la programación como una actividad fundamental en la educación básica, situada dentro de las competencias requeridas para el desarrollo de futuros profesionales que necesita ser evaluada.

Evaluar el pensamiento computacional es una cuestión crítica que ha motivado una amplia gama de investigaciones. En general, la mayoría de los estudios comienzan con una definición base de pensamiento computacional y construyen un marco teórico que incorpora dimensiones, principios, habilidades y actitudes, entre otros aspectos. Este marco sirve como base para desarrollar metodologías que permitan evaluar el pensamiento computacional de manera efectiva. Es notable que una gran cantidad de estas investigaciones se centran en poblaciones de educación primaria y secundaria, y menos en el nivel universitario. Además, estas investigaciones suelen referenciar herramientas específicas como Scratch, Alice, Kodu, Greenfoot, Agentsheets y Agentcubes. Por ejemplo, en el contexto de Scratch, Brennan y Resnick [9] desarrollan un marco conceptual compuesto por:

- 1. Conceptos:** las nociones fundamentales que se adquieren durante el aprendizaje del pensamiento computacional, generalmente incluyen secuencias, paralelismo, bucles, condicionales, eventos y operaciones de datos.
- 2. Prácticas computacionales:** se refiere al desarrollo de procedimientos que aplican conceptos computacionales para resolver problemas, centrándose en procesos de pensamiento y aprendizaje. Estos autores resaltan cuatro tipos principales de prácticas computacionales: desarrollo incremental e iterativo, depuración, reutilización y remezcla de soluciones, y abstracción y modularización.
- 3. Perspectivas computacionales:** entienden que resolver problemas computacionales fomenta una evolución en la perspectiva sobre tecnologías digitales, ayudando a las personas a ser más críticas, eficientes y estructuradas en la resolución de problemas y en la construcción de criterios para tomar decisiones. Destacan tres perspectivas principales: expresión, conexión y cuestionamiento.

Claramente, la propuesta de Brennan y Resnick se enfoca en el desarrollo gradual de habilidades específicas hacia habilidades más generales aplicables en la vida cotidiana. En lo que respecta a la evaluación del pensamiento computacional, subrayan la necesidad de emplear múltiples herramientas o estrategias, proponiendo el análisis de portafolios de proyectos, entrevistas basadas en proyectos y diseño de escenarios como enfoques complementarios.

Otro grupo de investigadores [17] ha presentado un test de pensamiento computacional, también basado en Scratch, dirigido a estudiantes de primaria y primeros niveles de secundaria. Este test evalúa aspectos como conceptos computacionales, entorno-interfaz del ítem, estilo de alternativas de respuesta, presencia de animaciones y la tarea requerida.

En 2015, Moreno-León y colaboradores [18] analizaron Dr. Scratch, una herramienta web gratuita y de código abierto que evalúa proyectos de Scratch, proporcionando una puntuación automática de pensamiento computacional y detectando posibles errores o malas prácticas de programación. Dr. Scratch se beneficia de plugins desarrollados por Hairball, una herramienta de análisis de código estático. Dr. Scratch evalúa competencias en siete conceptos clave del pensamiento computacional y define cuatro niveles de competencia: nulo, básico, en desarrollo y competente.

En 2019, Roman-González, Moreno-León y Robles [18] realizaron un diagnóstico de las herramientas disponibles para evaluar el pensamiento computacional, clasificándolas en categorías como herramientas de diagnóstico, sumativas, interactivas formativas, de minería de datos, para evaluar vocabulario, de transferencia de habilidades y escalas de percepción de actitudes. Aunque la mayoría de la investigación se centra en niveles de educación primaria y secundaria, un estudio reciente [15] busca abordar la evaluación del pensamiento computacional desde una perspectiva más amplia, considerando habilidades digitales definidas en el Foro Económico Mundial de 2018.

Este enfoque abarca tanto habilidades cognitivas como aspectos afectivos, destacando que el pensamiento computacional no solo es una actividad cognitiva, sino que también incluye elementos afectivos como actitudes y creencias sobre los problemas, dominio del problema y habilidades individuales para resolverlos. La investigación subraya la importancia de actitudes como la confianza, la persistencia, la tolerancia a la ambigüedad, la capacidad de comunicación y colaboración, así como la inteligencia espiritual, entre otras. Además, se enfoca en un conjunto de conceptos relacionados con la programación y, después de un proceso de validación riguroso, define elementos clave para la evaluación del pensamiento computacional, estructurados en tres dimensiones: habilidades cognitivas, actitudes e inteligencia espiritual.

En ese sentido, los investigadores han identificado un conjunto de conceptos relevantes, previamente mencionados, que están directamente relacionados con la programación, incluyendo:

- a) **Abstracción:** la habilidad de simplificar la complejidad enfocándose en la esencia.
- b) **Pensamiento algorítmico:** la competencia para secuenciar instrucciones de manera lógica.
- c) **Automatización:** la capacidad de hacer que procesos o tareas se ejecuten automáticamente.
- d) **Descomposición:** el proceso de dividir un problema complejo en partes más manejables.
- e) **Depuración:** el arte de detectar y corregir errores en el código o algoritmos.
- f) **Evaluación:** la capacidad de juzgar la efectividad o eficacia de diferentes soluciones.
- g) **Generalización:** la habilidad de aplicar soluciones a un rango más amplio de problemas.
- h) **Resolución de problemas:** la competencia de encontrar soluciones a problemas complicados.

- i) **Habilidad comunicativa:** la capacidad de transmitir y recibir información de manera efectiva.
- j) **Habilidad espiritual:** la aptitud para reflexionar y encontrar significado más allá de lo tangible.

Luego de un riguroso proceso de validación, los investigadores han definido los principales elementos a considerar en la evaluación del pensamiento computacional, estructurándolos en tres dimensiones:

a) Habilidades cognitivas, que incluyen:

- Abstracción: la habilidad de destilar un problema a su esencia, eliminando detalles innecesarios.
- Pensamiento algorítmico: la competencia para definir pasos claros y secuenciales para resolver problemas.
- Descomposición: la capacidad de fraccionar un problema complejo en subproblemas más simples.
- Depuración: la habilidad para identificar y corregir errores en una solución propuesta.
- Evaluación: la capacidad para analizar soluciones y mejorarlas.
- Generalización: la habilidad para aplicar soluciones de problemas anteriores a nuevos problemas.

b) Actitudes, que abarcan:

- Resolución de problemas: la capacidad para abordar y solucionar dilemas complicados.
- Trabajo en equipo: la habilidad para colaborar y trabajar armoniosamente con otros.
- Comunicación: la competencia para intercambiar información de manera clara y efectiva.

c) Inteligencia espiritual, que incluye:

- Autoconciencia: el conocimiento de las propias fortalezas y debilidades.
- Integridad: la adherencia a principios morales en la interacción con otros.
- Razonamiento creativo: la habilidad para concebir soluciones innovadoras.

Estos elementos se han establecido como fundamentales en la evaluación de pensamiento computacional y se basan en una comprensión profunda de las capacidades cognitivas, las actitudes y la inteligencia espiritual, como se detalla en la investigación de Roman-González, Moreno-León y Robles.

En el contexto de la programación en Scratch, herramientas como Dr. Scratch, desarrolladas por Moreno-León y otros [18], permiten una evaluación automática de proyectos, destacando su capacidad para analizar y proporcionar retroalimentación sobre las habilidades de pensamiento computacional. Esta herramienta se enriquece con plugins como Hairball, que facilita el análisis estático del código, identificando problemas y sugiriendo mejoras. La arquitectura flexible de Hairball permite la inclusión de nuevas funcionalidades, haciendo de Dr. Scratch un instrumento valioso en la evaluación de las habilidades de pensamiento computacional.

El sistema de evaluación dinámica ponderada (Dynamic Weighted Evaluation System - DWES) propuesto por Chai et al. [12] representa otro avance significativo en este campo, al permitir la evaluación automática de proyectos en Scratch y ajustar los criterios de evaluación en función

de las características específicas de cada proyecto. Este enfoque garantiza que los criterios de evaluación reflejen de manera más precisa las habilidades de pensamiento computacional manifestadas en los proyectos.

Además, investigaciones como la de Zhang et al. [12] subrayan la importancia de desarrollar indicadores de evaluación científicamente válidos que abarquen todas las dimensiones del pensamiento computacional. Estos indicadores son fundamentales para una evaluación precisa de las habilidades de pensamiento computacional y son esenciales para fomentar el desarrollo de estas habilidades. La validación de estos indicadores a través de análisis cualitativos, cuantitativos y evaluaciones de expertos asegura que reflejen de manera integral las capacidades de pensamiento computacional de los estudiantes.

Finalmente, como parte de los principales resultados obtenidos de este trabajo, en el Cuadro 2, se ofrece un resumen de los conceptos que se han mencionado con mayor frecuencia en la literatura consultada como representativos de la definición de pensamiento computacional. En menor medida, también se mencionan conceptos como codificación, automatización, control de flujo, diseño, interactividad, procesamiento de datos, modelo, control de flujo, programa, secuencias y transformación.

Cuadro 2. Frecuencia de conceptos asociados al pensamiento computacional.

Concepto	Ocurrencia
Abstracción	10
Pensamiento Algorítmico	8
Descomposición	8
Representación de datos	5
Depuración	4
Evaluación	4
Generalización	4
Paralelismo	4

Conclusiones

Este artículo representa un paso decisivo hacia la consolidación de un marco conceptual para el pensamiento computacional, subrayando su creciente importancia en la educación contemporánea y el desarrollo profesional. A través de una investigación rigurosa y un análisis detallado de estudios existentes, se ha logrado una comprensión integral del pensamiento computacional, destacando tanto las habilidades fundamentales como los procesos cognitivos asociados y su aplicabilidad en diversos contextos educativos y profesionales.

La contribución principal de este estudio radica en su capacidad para fusionar teorías conceptuales con aplicaciones prácticas, sirviendo de puente entre la teoría y la praxis. El marco conceptual establecido sienta las bases para currículos educativos avanzados y programas de capacitación, enfocándose en el desarrollo estructurado de habilidades esenciales en pensamiento computacional. Este enfoque es crucial para preparar a individuos para los retos de un mundo tecnológicamente avanzado, enfatizando la importancia de comprender, diseñar y optimizar sistemas computacionales.

Un hallazgo fundamental de este estudio es la necesidad imperante de formular una base consensuada para investigaciones futuras en pensamiento computacional. Esto implica la definición universal del concepto, la identificación de sus dimensiones y elementos

constituyentes, la creación de índices precisos, y el desarrollo de métricas objetivas para una evaluación efectiva y confiable del pensamiento computacional. Estos elementos son esenciales para avanzar de manera coherente y estructurada en este campo.

Tomando como base los conceptos más frecuentemente asociados al pensamiento computacional, se considera que los principales criterios para la subsecuente evaluación pueden enfocarse en: abstracción, pensamiento algorítmico, descomposición, representación de datos, depuración, evaluación, generalización y paralelismo.

Finalmente el artículo sienta las bases para futuras investigaciones en pensamiento computacional, ofreciendo un marco exhaustivo que sirve como punto de partida para investigaciones más profundas. La metodología detallada para la evaluación de habilidades computacionales, junto con la valoración de actitudes y la inteligencia espiritual en el aprendizaje, aporta una visión amplia y enriquecedora. Con este estudio, se abren caminos para futuras indagaciones y se proveen herramientas valiosas para educadores y profesionales, contribuyendo significativamente a la evolución y aplicación del pensamiento computacional en diversas disciplinas.

Agradecimientos

Este artículo es fruto del proyecto de investigación titulado “Metodología para el reconocimiento automático de patrones del Pensamiento Computacional en estudiantes de la educación general básica para mejorar los procesos de gestión”, desarrollado bajo el auspicio de la Vicerrectoría de Investigación y Extensión del Instituto Tecnológico de Costa Rica. Extendemos nuestro sincero agradecimiento a nuestra institución por su invaluable apoyo y por proporcionar los recursos necesarios para llevar a cabo esta importante iniciativa de investigación. Su compromiso con el fomento de la investigación científica y tecnológica ha sido fundamental para el éxito de este trabajo.

Referencias

- [1] J. M. Wing, “Computational thinking,” *Commun. ACM*, vol. 49, no. 3, pp. 33–35, Mar. 2006.
- [2] J. Cuny, L. Snyder, and J. Wing, “Demystifying computational thinking for non-computer scientists, work in progress,” 2010.
- [3] D. Seehorn, S. Carey, B. Fuschetto, I. Lee, D. Moix, D. O’Grady-Cunniff, ..., A. Verno, “CSTA K-12 Computer Science Standards: Revised 2011,” ACM, 2011.
- [4] M. Román-González, “Test de pensamiento computacional: principios de diseño, validación de contenido y análisis de ítems computational thinking test: design guidelines, content validation and item analysis.”
- [5] I. Dimos, C. Velaora, and A. Kakarountas, “Computational thinking in greek educational system for k-12: towards the future teaching approach,” in 2022 Panhellenic Conference on Electronics and Telecommunications (PACET), 2022, pp. 1–6. doi: 10.1109/PACET56979.2022.9976359.
- [6] S. Grover and R. Pea, “Computational thinking in k-12: A review of the state of the field,” *Educational researcher*, vol. 42, no. 1, pp. 38–43, 2013.
- [7] S. Bocconi, A. Chioccariello, G. Dettori, A. Ferrari, and K. Engelhardt, “El pensamiento computacional en la enseñanza obligatoria (computhink). implicaciones para la política y la práctica. developing computational thinking in compulsory education. implications for policy and practice,” 2016.
- [8] L. Seiter and B. Foreman, “Modeling the learning progressions of computational thinking of primary grade students,” in Proceedings of the ninth annual international ACM conference on International computing education research, 2013, pp. 59–66.
- [9] M. Resnick and K. Brennan, “Nuevos marcos de referencia para estudiar y evaluar el desarrollo del pensamiento computacional,” 2012.
- [10] X. Chai, Y. Sun, H. Luo, and M. Guizani, “Dwes: A dynamic weighted evaluation system for scratch based on computational thinking,” *IEEE Transactions on Emerging Topics in Computing*, vol. 10, no. 2, pp. 917–932, 2022. doi: 10.1109/TETC.2020.3044588.

- [11] S. Rijo-García, E. Segredo, and C. León, "Computational thinking and user interfaces: A systematic review," *IEEE Transactions on Education*, vol. 65, no. 4, pp. 647–656, 2022. doi: 10.1109/TE.2022.3159765.
- [12] W. Zhang, X. Zeng, D. Ming, and J. Wang, "Research on the construction of evaluation indicators of students' computational thinking based on spectral clustering," in *2022 10th International Conference on Information and Education Technology (ICIET)*, 2022, pp. 104–112. doi: 10.1109/ICIET55102.2022.9779003.
- [13] F. Luo, M. Israel, and B. Gane, "Elementary computational thinking instruction and assessment: A learning trajectory perspective," *ACM Trans. Comput. Educ.*, vol. 22, no. 2, Feb. 2022. doi: 10.1145/3494579.
- [14] R. Ismail, T. A. Steinbach, and C. S. Miller, "A guide towards a definition of computational thinking in k-12," in *2022 IEEE Global Engineering Education Conference (EDUCON)*, 2022, pp. 801–810. doi: 10.1109/EDUCON52537.2022.9766703.
- [15] D. E. Sondakh, K. Osman, and S. Zainudin, "A proposal for holistic assessment of computational thinking for undergraduate: Content validity," *European Journal of Educational Research*, vol. 9, no. 1, pp. 33–50, 2020.
- [16] R. Barrera Capot, R. Montaño Espinoza, and J. Sánchez, "Desarrollo del pensamiento computacional con scratch," *Nuevas Ideas en Informática Educativa*, vol. 11, pp. 616–620, 2015.
- [17] M. Román-González, J. C. Pérez-González, and C. Jiménez-Fernández, "Test de pensamiento computacional: diseño y psicometría general," in *III Congreso Internacional sobre Aprendizaje, Innovación y Competitividad (CINAIC 2015)*, 2015, pp. 1–6.
- [18] M. Román-González, J. Moreno-León, and G. Robles, "Combining assessment tools for a comprehensive evaluation of computational thinking interventions," in *Computational Thinking Education*, 2019, pp. 79–98.
- [19] B. Boe, C. Hill, M. Len, G. Dreschler, P. Conrad, and D. Franklin, "Hairball: Lint-inspired static analysis of Scratch projects," in *Proceedings of the 44th ACM Technical Symposium on Computer Science Education*, 2013, pp. 215–220.

Declaración sobre uso de Inteligencia Artificial (IA)

Los autores declaramos que hemos utilizado una herramienta de inteligencia artificial ChatGPT para asistirnos en la redacción de este artículo. Esta herramienta nos ayudó a mejorar la estructura y la claridad del texto. Los contenidos generados por la IA fueron revisados minuciosamente por nosotros para asegurar su precisión y coherencia con el objetivo del estudio.