



Universidad  
Internacional  
de Andalucía

## TÍTULO

**EL USO DEL LENGUAJE DE PROGRAMACIÓN PYTHON EN LA  
ENSEÑANZA Y APRENDIZAJE DE LAS MATEMÁTICAS**  
UNA REVISIÓN SISTEMÁTICA DE LA LITERATURA

## AUTOR

**Camilo Montoya Puerta**

**Esta edición electrónica ha sido realizada en 2026**

Tutor	Dr. Luis Carlos Contreras González
Instituciones	Universidad Internacional de Andalucía; Universidad de Huelva
Curso	<i>Máster Universitario en Investigación en la Enseñanza y el Aprendizaje de las Ciencias Experimentales, Sociales y Matemáticas (2024/25)</i>
©	Camilo Montoya Puerta
©	De esta edición: Universidad Internacional de Andalucía
Fecha documento	2025



Universidad  
Internacional  
de Andalucía



**Atribución-NoComercial-SinDerivadas  
4.0 Internacional (CC BY-NC-ND 4.0)**

Para más información:

<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>

<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.en>



Universidad  
de Huelva

**El Uso Del Lenguaje De Programación Python En La Enseñanza Y Aprendizaje De Las  
Matemáticas: Una Revisión Sistemática De La Literatura**

**Alumno**

Camilo Montoya-Puerta

Trabajo de Fin de Máster

Artículo de investigación

**Asesor**

Dr. Luis Carlos Contratas Gonzáles

Máster Universitario en Investigación en la Enseñanza y el Aprendizaje de las Ciencias

Experimentales, Sociales y Matemáticas

Universidad de Huelva

Curso 2024-2025

## Tabla de contenido

Resumen.....	3
Abstract.....	4
Introducción.....	5
Objetivos.....	7
Objetivo general.....	7
Objetivos específicos.....	7
Fundamentación teórica.....	7
La Programación y su relación con las Matemáticas.....	7
Los Lenguajes de Programación en la Educación Matemática.....	8
El Pensamiento Computacional.....	9
Metodología.....	10
Estrategia de búsqueda y selección.....	10
Sistema de Categorías.....	11
Resultados.....	13
Uso de Python en la Enseñanza y Aprendizaje de las Matemáticas.....	14
Tendencias y Enfoques Metodológicos.....	16
Beneficios y Limitaciones del Vínculo entre Python y las Matemáticas.....	17
Discusión.....	19
Conclusiones.....	20
Referencias.....	22

## Lista de tablas

<b>Tabla 1.</b> Sistema de categorías para el análisis de la literatura.....	12
--	----

## Lista de figuras

<b>Figura 1.</b> <i>Diagrama de flujo.</i> .....	13
--	----

## Resumen

Esta investigación se centra en la integración de la programación en la educación matemática, destacando el lenguaje Python por su potencial para despertar el interés de los estudiantes, así como por su simplicidad, versatilidad y capacidad para facilitar la comprensión conceptual de las matemáticas. En este sentido, se propuso como objetivo general conocer los usos de este lenguaje de programación en la enseñanza y aprendizaje de las matemáticas a partir de una revisión sistemática de la literatura, la cual fue guiada por el protocolo PRISMA. En esta se seleccionaron 19 estudios encontrados en bases de datos de alto impacto, y la información recolectada fue organizada a través de un sistema de categorías construido a partir de diferentes fundamentos teóricos.

Estos datos evidenciaron que Python se emplea en la resolución de problemas matemáticos, el diseño y desarrollo de cursos, la construcción de recursos digitales y en proyectos interdisciplinarios, con especial presencia en contextos universitarios. Además, se expuso que el uso de este lenguaje favorece el tránsito de lo abstracto a lo concreto y manejable, lo que promueve la comprensión, motivación, autonomía e inclusión en los estudiantes. No obstante, aún persisten algunas limitaciones vinculadas a la falta de formación docente, la desigualdad en el acceso a los recursos y la escasa incorporación de Python en niveles escolares iniciales. A su vez, esta revisión muestra que Python es una herramienta pedagógica eficaz para vincularse con las matemáticas y ofrecer experiencias significativas, pero su implementación requiere de apoyos institucionales y de políticas educativas que aseguren la sostenibilidad y equidad educativa.

*Palabras clave:* Python, educación matemática, pensamiento computacional y matemático, lenguajes de programación.

### **Abstract**

This research focuses on the integration of programming into mathematics education, highlighting the Python language for its potential to spark students' interest, as well as for its simplicity, versatility, and ability to facilitate the conceptual understanding of mathematics. In this regard, the general objective was to examine the uses of this programming language in the teaching and learning of mathematics through a systematic literature review guided by the PRISMA protocol. Nineteen studies from high-impact databases were selected, and the collected information was organized using a system of categories built on various theoretical foundations.

The findings revealed that Python is employed in mathematical problem-solving, course design and development, the creation of digital resources, and interdisciplinary projects, with a particularly strong presence in university contexts. Moreover, it was shown that the use of this language supports the transition from the abstract to the concrete and manageable, thereby fostering understanding, motivation, autonomy, and inclusion. However, some limitations remain, such as insufficient teacher training, inequality in access to resources, and the limited incorporation of Python in early school levels. At the same time, this review demonstrates that Python is an effective pedagogical tool for engaging with mathematics and providing meaningful learning experiences, though its implementation requires institutional support and educational policies that ensure sustainability and equity in education.

*Keywords:* Python, mathematics education, computational and mathematical thinking, programming languages.

## Introducción

“Dejemos de tratar de hacer que a los niños les gusten las matemáticas que odian, hagamos una matemática que ellos amen”  
Seymour Papert.

La incorporación de herramientas digitales y el desarrollo del pensamiento computacional (PC) en las aulas han transformado los procesos de enseñanza y aprendizaje de las diferentes disciplinas. En particular, la didáctica de las matemáticas se ha enfocado en indagar sobre estos cambios, explorando nuevas metodologías y recursos para mejorar la comprensión de los conceptos matemáticos. Un claro ejemplo de esto es el uso de la programación como recurso didáctico en la enseñanza de las matemáticas, la cual facilita un entorno interactivo donde los estudiantes pueden aprender y aplicar sus conocimientos de manera práctica.

Desde la propuesta del lenguaje de programación Logo, diseñado por Seymour Papert a finales de la década de los 60 's, se ha introducido la programación en la educación. Papert fue uno de los pioneros en entablar el vínculo entre esta disciplina y las matemáticas, diseñando Logo especialmente para favorecer la comprensión de conceptos matemáticos y de programación en los estudiantes más jóvenes (Gillespie, 2004). Logo permitía a los alumnos interactuar con la computadora mediante una serie de instrucciones dirigidas a una “tortuga”, con el fin de generar figuras a partir de su movimiento, lo cual facilitaba el aprendizaje de la geometría y la lógica algorítmica (Gillespie, 2004). Iniciativas como esta han ayudado al posicionamiento del uso del PC en las escuelas y, desde entonces, la programación ha sido vista como una herramienta que no solo posibilita la resolución de problemas, sino que también, fomenta la creatividad, la imaginación, la lógica (Rivera Muñiz, 2021), el aprendizaje basado en experimentación (De la Hoz y Hijón, 2022) y la adquisición de habilidades tanto para razonar sistemáticamente como para trabajar de manera colaborativa (Resnick et al., 2009).

Papert (1980), más allá de promover el vínculo, defendió la idea de que el aprendizaje de las matemáticas debería ir más allá de la simple memorización de fórmulas y procedimientos, y sugirió que la programación ofrece un medio ideal para fomentar el aprendizaje activo, donde los estudiantes pueden acercarse a conceptos matemáticos a través de la creación de sus propios proyectos. A partir de entonces, este enfoque ha evolucionado y, hoy en día, el uso de los lenguajes de programación se considera como un recurso de gran valor en el aula, no solo para enseñar programación en cuanto tal, sino también para mejorar la comprensión de las matemáticas.

En particular, Python se ha consolidado como uno de los lenguajes favoritos en la enseñanza debido a su facilidad de aprendizaje (Auccahuasi et al., 2018), claridad y versatilidad para modelar problemas matemáticos. Sus bibliotecas, como Numpy o Sympy, ofrecen una amplia colección de funciones y algoritmos, las cuales resultan de gran utilidad para el desarrollo del pensamiento matemático a mayor escala (Betteridge et al., 2022). Así pues, la programación, y en particular lenguajes como Python, no solo contribuyen a mejorar la comprensión de las matemáticas, sino que también permiten a los estudiantes desarrollar habilidades fundamentales para responder a las demandas actuales de la sociedad. Esto se debe a que la programación facilita la adquisición de competencias educativas y digitales, a la vez que promueve el uso adecuado de las tecnologías de la información y la comunicación (De la Hoz y Hijón, 2022). Sin embargo, a pesar del potencial de la programación en la enseñanza de las matemáticas, persisten barreras para su incorporación en las aulas.

En diferentes contextos educativos, la enseñanza de la programación se ha limitado al aprendizaje de su sintaxis (Palma y Sarmiento, 2015), sin lograr una integración significativa con

otras disciplinas del conocimiento, lo que implica desaprovechar gran parte de su potencial y posibilidades. Además, muchos docentes aún enfrentan dificultades debido a su falta de formación en programación, incluso algunos la perciben como un obstáculo para el desarrollo de sus procesos de enseñanza (Kaufmann et al., 2023). Esto evidencia que, además, de la falta de formación, algunos docentes muestran resistencia para cambiar las metodologías de aula tradicionales, las cuales, en la mayoría de los casos proporcionan un aprendizaje lento de las asignaturas (Auccahuasi et al., 2018). Esta situación dificulta la implementación de propuestas diferentes e innovadoras y conlleva a la baja motivación en los estudiantes al momento de aprender matemáticas, una disciplina que perciben como abstracta y compleja, en línea de lo que Papert (1980) señalaba como *mathfobia*.

Sumado a esto, los currículos educativos de países como Colombia o México no promueven la integración de la programación en la enseñanza de las matemáticas, lo cual limita la motivación de los docentes para cambiar su manera de llevar a cabo los procesos de enseñanza y aprendizaje. En contraste, en países como Noruega, Inglaterra, Estados Unidos y Canadá, la programación forma parte de los planes de estudio de matemáticas (Kaufmann et al., 2023), incluso desde los primeros niveles educativos (Palma y Sarmiento, 2015). No obstante, a pesar de estos desafíos, la integración de la programación en la enseñanza de las matemáticas representa una oportunidad para superarlos. Para lograrlo se debe entonces promover su desarrollo en las aulas, ya que los lenguajes de programación requieren de conocimientos matemáticos (Auccahuasi et al., 2018) y, al mismo tiempo, programar constituye una forma eficaz de aprender matemáticas (Betteridge et al., 2022).

En este sentido, la integración de Python en la enseñanza y aprendizaje de las matemáticas constituye una alternativa frente a las metodologías de enseñanza tradicionales, al presentar la disciplina de una forma más atractiva. Esto puede contribuir a motivar a los estudiantes ya que les facilita demostrar y aplicar sus conocimientos (Betteridge et al., 2022), a la vez que les ayuda a encontrar un sentido a las matemáticas, con una visión más profunda y simplificada (Rivera Muñoz, 2021), lo que facilita abordar problemas complejos de una manera sencilla. Asimismo, su incorporación en las aulas impulsa a los docentes a formarse en programación con el fin de responder a las demandas sociales actuales y a las de sus estudiantes, contribuyendo así a la formación de personas críticas y competentes (Betteridge et al., 2022).

Ahora bien, autores como Tejera-Martínez et al. (2020), afirman que, en el campo investigativo, se evidencia una carencia en estudios sobre la aplicación de lenguajes de programación en las aulas, sobre todo aquellos que potencien la motivación de los estudiantes hacia el aprendizaje de las matemáticas (De la Hoz y Hijón, 2022). En este sentido, el propósito de esta investigación es realizar una revisión sistemática de la literatura con el fin de describir la evidencia disponible sobre el uso de Python en la enseñanza y aprendizaje de las matemáticas, considerando su potencial y capacidad de generar interés en los estudiantes. Esta revisión busca responder la pregunta de investigación  $Q_0$ : ¿Qué uso se le ha dado al lenguaje de programación Python en la enseñanza y aprendizaje de las matemáticas en diferentes niveles educativos, según la literatura existente?

Para cumplir dicho propósito, se realizará una búsqueda en bases de datos reconocidas en el campo educativo, aplicando criterios específicos de inclusión y exclusión para seleccionar estudios relevantes sobre el tema. La bibliografía encontrada se organizará y analizará en función del uso de Python, de las metodologías o enfoques empleados y los resultados, como las limitaciones y beneficios observados, con el fin de ofrecer una visión integral de la literatura. Por

último, se presentarán algunas consideraciones a partir de la información encontrada en esta revisión.

## Objetivos

Los objetivos de esta investigación se estructuran en dos niveles, los cuales se presentan de manera clara y detallada para garantizar precisión en el proceso investigativo. En el primer nivel se plantea un objetivo general que define el propósito central de la investigación y, en el segundo nivel, se establecen los objetivos específicos que permiten abordar aspectos claves del análisis (González, 2011).

### Objetivo general

Conocer los usos del lenguaje de programación Python en la enseñanza y aprendizaje de las matemáticas, a partir de una revisión sistemática de la literatura.

### Objetivos específicos

Una vez planteado el objetivo general, se proponen los siguientes objetivos específicos para alcanzarlo:

- i. Identificar las principales tendencias o enfoques metodológicos en los estudios que aborden el uso de Python en la enseñanza de las matemáticas.
- ii. Describir los beneficios y limitaciones identificados en la literatura sobre la integración de Python en los procesos de enseñanza y aprendizaje matemático.
- iii. Definir cómo a partir de la literatura existente la programación con Python contribuye al desarrollo del pensamiento matemático y computacional.

El cumplimiento de estos objetivos permitirá obtener una visión amplia sobre el uso de Python en la enseñanza de las matemáticas, proporcionando insumos para futuras investigaciones. Además, se espera fortalecer el vínculo entre la programación y las matemáticas, promoviendo enfoques que potencien la motivación, el pensamiento matemático y el PC en los estudiantes.

## Fundamentación teórica

En este apartado se presentan los fundamentos teóricos de esta investigación, los cuales están divididos en tres ejes centrales. En el primero, se aborda la programación y su relación con las matemáticas, donde se describe lo que se entiende por tal concepto y cómo contribuye al desarrollo del pensamiento matemático. En el segundo, se habla de los lenguajes de programación en la educación matemática y cómo sus usos favorecen la comprensión de conceptos abstractos. En el tercero, se aborda el pensamiento computacional, entendido como una competencia clave en la formación matemática.

### La Programación y su relación con las Matemáticas

La programación ha adquirido un papel relevante en ámbitos como las ciencias, la ingeniería, la informática y las matemáticas (Wing, 2008). En el campo educativo, su incorporación en los procesos de enseñanza no solo fomenta las competencias digitales, sino que

también, facilita la comprensión y el abordaje de los conocimientos en diferentes disciplinas. De acuerdo con Kaufmann et al. (2023), la programación puede ser entendida como un conjunto de instrucciones secuenciales diseñadas para que una computadora o un dispositivo digital ejecute tareas o resuelva problemas, lo cual implica una estructuración lógica y un pensamiento ordenado.

En este sentido, la programación puede concebirse como una herramienta que no solo asiste trabajos mecánicos y sistemáticos, sino que también, apoya tareas cognitivas relacionadas con el PC y posibilita demostrar competencias computacionales (Grover y Pea, 2013). Además, la programación facilita el diseño de soluciones, la generación ideas y la aplicación de la creatividad en sus procesos algorítmicos, a la vez que favorece la adquisición de destrezas vinculadas al pensamiento matemático (De la Hoz y Hijón, 2022).

Esto cobra mayor significado al considerar que el pensamiento matemático, de acuerdo con los planteamientos expuestos por Bosch Saldaña (2021), es una manera dinámica de pensar que integra habilidades como la abstracción, la visualización y la modelización. Dicho pensamiento no se limita al ámbito científico, sino que está presente en la vida cotidiana al enfrentar diversas tareas, incluidas aquellas que se realizan en dispositivos digitales. Tales tareas pueden ser fortalecidas por la programación, con el fin de entablar una relación con procesos matemáticos. Asimismo, desde esta perspectiva, se integran aspectos emocionales y motivacionales que influyen en el desarrollo y aplicación de ideas matemáticas.

Propuestas como la de Papert (1980) refuerzan estos planteamientos, pues evidencian que la programación no consiste simplemente en dar órdenes a una máquina, sino que constituye un medio para construir conocimientos y pensar matemáticamente a partir de la acción. En esta misma línea, autores como Aucahuasi et al. (2018) destacan que el dominio de la programación requiere de conocimientos en matemáticas, lo que evidencia un vínculo inherente entre ambas disciplinas. Además, cabe señalar que, a nivel estructural, la programación se basa en abstracciones y estructuras formales propias del pensamiento matemático, ya que la informática, al igual que muchas otras ciencias, se fundamenta en principios matemáticos (Wing, 2006).

Así pues, al compartir estructuras conceptuales con las matemáticas, la programación constituye una vía pertinente para facilitar su aprendizaje. Su integración en el aula favorece una comprensión profunda, aplicada y significativa de los contenidos, al tiempo que posibilita el desarrollo de capacidades como el razonamiento lógico, el pensamiento estructurado y la resolución de problemas. En este sentido, la programación se convierte en una estrategia efectiva para fortalecer el pensamiento matemático y aprender conceptos desde una perspectiva práctica (Pinargote-Zambrano et al., 2024).

### **Los Lenguajes de Programación en la Educación Matemática.**

Desde un punto de vista teórico, los lenguajes de programación pueden considerarse como un sistema de abstracción que transforma algoritmos; es decir, convierten procedimientos secuenciales de entrada en operaciones computables de salida (Wing, 2008). Esta característica ha permitido su diversificación en función de sus aplicaciones en diferentes campos. En la educación, con la aparición del lenguaje Logo, han surgido gran cantidad de apuestas que permiten integrar la tecnología en los procesos de enseñanza y aprendizaje de las matemáticas (De la Hoz y Hijón, 2022).

Un ejemplo inspirado en Logo es el lenguaje SET Interactivo, el cual posibilita la asignación de instrucciones a objetos y variables para interactuar y experimentar a través del uso del lenguaje. Este también ha sido implementado para apoyar el aprendizaje matemático, especialmente desde etapas tempranas (De la Hoz y Hijón, 2022), ya que puede aplicarse en

diversas dinámicas como juegos o situaciones didácticas. En esta misma perspectiva, según De la Hoz e Hijón (2022), Scratch es un lenguaje que se ha consolidado como uno de los más utilizados y estudiado en entornos escolares, esto debido a su metodología de programación por bloques, su carácter visual y su adaptabilidad a las distintas etapas formativas. El uso de este lenguaje promueve el aprendizaje significativo, estimula el pensamiento computacional y facilita el desarrollo de competencias matemáticas (Tejera-Martínez et al., 2020).

En relación con la educación matemática en niveles superiores, lenguajes como Matlab, Maple, Mathematica y Python, se han convertido en una herramienta para abordar los contenidos matemáticos (Betteridge et al., 2022). Estos no solo permiten reducir los tiempos de desarrollo y visualización de procesos, sino que también, amplían las posibilidades de representación y análisis matemático (Betteridge et al., 2022). En particular, Python, como se ha señalado anteriormente, es uno de los lenguajes preferidos por estudiantes y docentes, no solo por su funcionalidad técnica, sino también, por su potencial para desarrollar proyectos, su simpleza y claridad en su aprendizaje (Auccahuasi et al., 2018). Este lenguaje cuenta con múltiples librerías para el procesamiento y análisis de datos, que a su vez se emplean tanto para realizar operaciones matemáticas avanzadas como para desarrollar y entrenar inteligencias artificiales.

En este sentido, Python posee un gran potencial y es utilizado en la enseñanza como una estrategia didáctica que permite desarrollar actividades interactivas, realizar simulaciones, visualizar datos y trabajar en proyectos aplicados, los cuales promueven la motivación y el aprendizaje activo y contextualizado (Pinargote-Zambrano et al., 2024). Además, su aplicación en áreas como el álgebra, el cálculo, la estadística y el análisis, favorece la comprensión de conceptos abstractos a través de experiencias prácticas (Pinargote-Zambrano et al., 2024), posibilidades que refuerzan su valor pedagógico en la enseñanza de las matemáticas. En consecuencia, Python constituye un medio eficaz para acercar a los estudiantes al pensamiento matemático desde una perspectiva integradora, vinculada a los desafíos y demandas de los contextos actuales.

## **El Pensamiento Computacional**

El pensamiento computacional (PC) ha adquirido relevancia en las últimas décadas como una competencia clave en el contexto educativo, especialmente, por su potencial para fortalecer el aprendizaje de disciplinas como las matemáticas. Esta noción fue introducida por Seymour Papert (1980), quien la vinculó con la programación y el desarrollo del pensamiento lógico-matemático en los contextos escolares. Sin embargo, fue Jeannette Wing quien desarrolló el concepto como tal, dándole visibilidad y estructura en el campo investigativo. Así, Según Wing (2006), “el pensamiento computacional implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, recurriendo a los conceptos fundamentales de la ciencia de la computación” (p. 33).

Las ideas expuestas por Wing despertaron el interés en la comunidad académica impulsando múltiples investigaciones que consideran el PC como un campo de estudio dentro de la educación (Grover y Pea, 2013). Desde entonces, el PC ha sido objeto de análisis por parte de investigadores, educadores y responsables de las políticas educativas, lo que ha permitido expandir sus alcances teóricos y didácticos, así como su integración en algunos currículums escolares. En este sentido, el PC adquirió un enfoque más amplio, incorporando apuestas como el modelado, la descomposición de problemas y la planificación de soluciones sistemáticas (Grover y Pea, 2013; Wing, 2008).

Wing (2008) señala que la esencia del PC es la abstracción, es decir, la capacidad de decidir qué detalles son relevantes para resolver un problema y cuáles pueden ser ignorados. En este

sentido, este tipo de pensamiento se basa en procesos como la descomposición de problemas abstractos y complejos en otros más manejables que puedan resolverse de manera sistemática. De este modo, la formulación de soluciones se lleva a cabo mediante algoritmos y generalizaciones para representar de manera eficiente los aspectos esenciales de un problema (Kaufmann et al., 2023; Wing, 2006). Esto posiciona al PC como herramienta cognitiva para abordar la resolución de problemas e, incluso, desafíos cada vez más mediados por la tecnología. Si bien en sus inicios estuvo estrechamente asociado con la programación y a la escritura de código, con el tiempo su alcance teórico se amplió, incorporando conceptos de las ciencias computacionales, como el modelado, la recursividad, la generalización y la representación de datos (Grover y Pea, 2013; Wing, 2008).

Esta ampliación ha dado lugar a múltiples interpretaciones sobre la definición y aplicación del PC, generando debates en el ámbito educativo e investigativo acerca de su implementación y su relación con diferentes disciplinas. En el caso de la didáctica de las matemáticas, el PC se concibe como un recurso para desarrollar habilidades como el razonamiento sistemático y la resolución de problemas; asimismo, facilita que los estudiantes enfrenten desafíos matemáticos mediante estrategias como la identificación de patrones, formulación de algoritmos y modelización de soluciones.

Desde esta perspectiva, la implementación del PC en el aula promueve una actitud crítica y flexible ante los problemas, incentivando la búsqueda de soluciones creativas y el uso de herramientas tecnológicas para la construcción de conocimiento. Como lo señala Wing (2006), el PC no es una habilidad única de los informáticos, sino una forma de pensar que es aplicable a cualquier área del conocimiento y, por lo tanto, constituye una competencia fundamental en la actualidad para la formación del pensamiento matemático.

### **Metodología**

“Lo importante en la ciencia no es tanto obtener nuevos datos, sino descubrir nuevas formas de pensar sobre ellos”  
William Lawrence Bragg

En esta investigación se desarrolló una revisión sistemática de la literatura para recopilar, analizar y sintetizar información relevante sobre el uso del lenguaje de programación Python en la enseñanza y aprendizaje de las matemáticas. En esta revisión se identifican, seleccionan y examinan estudios primarios de manera estructurada (Marín, 2022) y se busca responder a una pregunta de investigación específica (en este caso  $Q_0$ ) la cual es planteada de manera explícita y sistemática (Ferreria et al., 2011). Para garantizar la rigurosidad y la transparencia en la recolección y análisis de los datos, se adoptó el protocolo PRISMA (Preferred Reporting Items For Systematic Reviews And Meta-Analyses), el cual es una guía reconocida y recomendada por su aplicación en revisiones sistemáticas (Marín, 2022; Sánchez Serrano et al., 2022). A continuación, se presenta la estrategia empleada en esta revisión para la recolección de datos.

### **Estrategia de búsqueda y selección**

La estrategia se fundamentó en la consulta de las bases de datos científicas de alto impacto Scopus y Web of Science (WoS), debido a su reconocimiento por la calidad de sus publicaciones en el campo de la investigación educativa. Para la recolección de información se diseñó una ecuación de búsqueda utilizando los términos clave combinados: Python AND education AND

mathematics\* AND (“learning” OR “teaching”), asegurando así una cobertura amplia y pertinente del tema de estudio. Posteriormente, para la selección de datos, se establecieron y aplicaron algunos criterios de inclusión y exclusión específicos.

Por un lado, con el fin de garantizar la actualidad de los estudios, se incluyeron aquellos que fueron publicados entre 2015 e inicios de 2025, periodo que corresponde al marco temporal en el cual se desarrolla esta investigación. Asimismo, se seleccionaron aquellos estudios que estuvieran escritos en español o inglés, ya que gran parte de las publicaciones de Scopus y WoS se encuentran en estos idiomas. Además, se estableció como criterio que existiera una relación clara entre el uso de Python y la enseñanza y aprendizaje de las matemáticas. También, se incluyeron investigaciones de tipo empírico y experimental, así como aportaciones presentadas en congresos que ofrecieran evidencia concreta sobre el uso de Python en entornos educativos.

Por otro lado, se excluyeron libros, tesis y estudios que no mencionaran en su título, resumen o palabras clave la relación entre Python y la educación matemática. Asimismo, no se tomaron en consideración aquellos que se centraban exclusivamente en la enseñanza de Python, sin establecer vínculos con procesos de enseñanza y aprendizaje de las matemáticas, así como aquellos que estuvieran duplicados en las bases de datos.

Se elaboró y utilizó como herramienta para organizar y visualizar este proceso de selección un diagrama de flujo propuesto por el protocolo PRISMA, el cual se basó en cuatro fases: identificación, cribado, elegibilidad e inclusión de la cantidad total de estudios para la revisión. Con el fin de organizar y sistematizar la información, se utilizaron apuntes y subrayados de lectura, y se elaboró una matriz de extracción de datos en Microsoft Excel que incluyó los siguientes elementos: base de datos, idioma, año, título, referencias y un espacio para observaciones.

El análisis se desarrolló a partir de un enfoque cualitativo situado en el paradigma interpretativo, el cual facilita comprender contextos, fenómenos y prácticas educativas (Cresswell, 2013), entre ellos los desafíos y oportunidades asociados a la integración de Python en los procesos de aprendizaje de las matemáticas, todo ello a partir de un sistema de categorías previamente construido. Este sistema se elaboró en dos versiones: la primera se construyó a partir de las ideas identificadas de autores como Papert (1980), Kaufmann et al. (2023), Wing (2006), entre otros, cuyos aportes teóricos sirvieron de base para fundamentar las categorías y subcategorías.

Por otro lado, la segunda versión se elaboró a partir de los comentarios y ajustes sugeridos a partir de una triangulación con expertos. Esto permitió consolidar un instrumento clave para organizar la información en torno al uso de Python, los enfoques metodológicos y los beneficios y limitaciones de los usos de este lenguaje de programación. Asimismo, esto contribuyó a establecer una base sólida para examinar las implicaciones pedagógicas del vínculo entre Python y la educación matemática. Finalmente, se extrajeron y se presentaron algunas conclusiones de este proceso fundamentadas en los resultados de esta revisión.

## **Sistema de Categorías**

El sistema de categorías se deriva de los fundamentos teóricos de esta investigación, como se mencionó anteriormente, y responde a la estructura organizativa planteada en la metodología (ver tabla 1). Para su construcción se establecieron tres categorías principales: el uso de Python en la enseñanza y aprendizaje de las matemáticas, las tendencias y enfoques metodológicos, y, por último, los beneficios y limitaciones del vínculo entre Python y las matemáticas. A cada categoría le corresponde sus respectivas subcategorías y descriptores, con el fin facilitar la organización e interpretación de la información de manera sistemática y coherente.

**Tabla 1.**

*Sistema de categorías para el análisis de la literatura.*

<b>Categoría</b>	<b>Subcategoría</b>	<b>Descriptor</b>
<b>Uso de Python en la enseñanza y aprendizaje de las matemáticas</b>	Aplicaciones en procesos matemáticos	Se emplea en simulaciones, modelado, visualización de datos o resolución de problemas (Betteridge et al., 2022; Pinargote-Zambrano et al., 2024).
	Áreas del conocimiento matemático	Se utiliza para el aprendizaje profundo o refuerzo matemático del álgebra, estadística, cálculo, etc (Betteridge et al., 2022).
	Finalidades pedagógicas	Se usa para diseñar recursos y aplicativos digitales o evaluar aprendizajes (Rivera-Muñiz, 2021).
	Nivel educativo	Primaria, secundaria, universidad, formación docente u otros (Auccahuasi et al., 2018).
<b>Tendencias y enfoque metodológicos</b>	Aprendizaje basado en proyectos (ABP), Educación STEAM, gamificación, aprendizaje activo, experimental o práctico y enfoques inclusivos	Corrientes metodológicas que fundamentan el uso de programación en educación matemática (de la Hoz Ruiz et al., 2022; Papert, 1980).
	Intervenciones pedagógicas, secuencias didácticas, planificación de cursos	Implementaciones concretas del uso de Python en las aulas como desarrollo de cursos (de la Hoz Ruiz et al., 2022).
	Enfoques tradicionales	Predominio de la memorización y aplicación mecánica de procedimientos matemáticos o de programación (Auccahuasi et al., 2018; Papert, 1980).
<b>Beneficios y limitaciones</b>	Beneficios pedagógicos y didácticos	Comprensión de contenido, vínculos interdisciplinarios, competencias digitales (Auccahuasi et al., 2018; Kaufmann et al., 2023).
	Condiciones contextuales desfavorables	Poca formación docente, falta de tiempo o recursos tecnológicos (Kaufmann et al., 2023; Palma et al., 2015).
	Obstáculos institucionales	Dificultades asociadas a la falta de integración en el currículo, resistencia a apuestas que generen este vínculo, dificultad en la sintaxis (Kaufmann et al., 2023).
	Desarrollo del pensamiento matemático	Mejora del razonamiento lógico, comprensión conceptual, resolución de problemas (Papert, 1980; Pinargote-Zambrano et al., 2024).
	Desarrollo del pensamiento computacional	Uso de algoritmos, abstracción, descomposición, modelos computacionales (Pea, 2013; Wing, 2006).
	Aspectos afectivos del aprendizaje	Motivación, interés, cambio de actitud hacia las matemáticas (Betteridge et al., 2022; de la Hoz Ruiz et al., 2022).

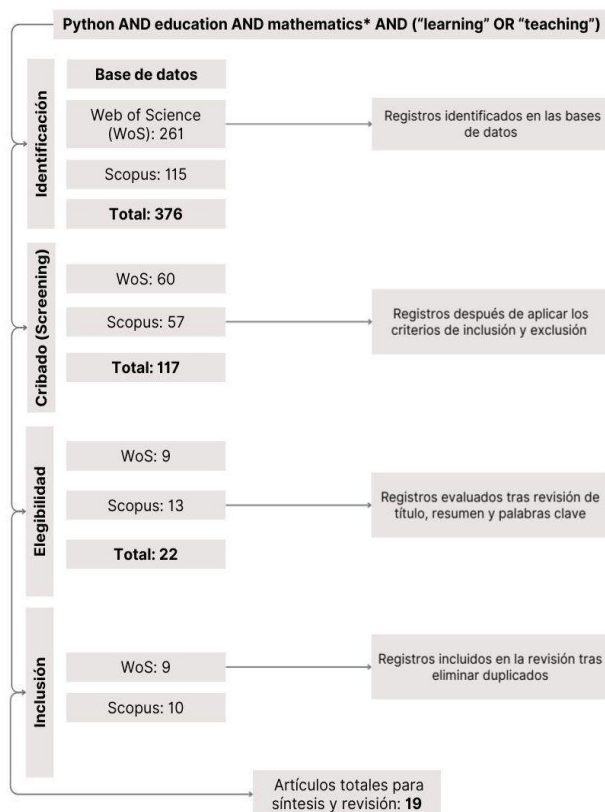
*Nota:* Elaboración propia.

Además, cada categoría se encuentra relacionada con los objetivos de la investigación. Así, el objetivo general, orientado a conocer los usos de Python en la enseñanza y aprendizaje de las matemáticas, se vincula con la categoría 1. A su vez, el objetivo específico orientado a identificar los enfoques se relaciona con la categoría 2. Finalmente, los objetivos que buscan describir los beneficios y limitaciones, así como el aporte de Python al desarrollo del pensamiento matemático y computacional, se corresponden con la categoría 3. De este modo, se garantiza que el análisis de la literatura esté alineado con el propósito de la investigación, lo que permite una lectura estructurada y fundamentada de los hallazgos.

## Resultados

Esta revisión sistemática de la literatura se desarrolló en cuatro fases, con base en el protocolo PRISMA (ver figura 1), lo que permitió analizar de manera rigurosa los estudios incluidos en la investigación.

**Figura 1.**  
*Diagrama de flujo.*



*Nota:* Diagrama basado en el protocolo PRISMA.

En la primera fase, de identificación, se recopilaron 376 registros provenientes de las dos bases de datos: 261 de WoS y 115 de Scopus. En la segunda fase, de cribado, se aplicaron los criterios de inclusión y exclusión previamente definidos, lo que redujo la muestra a 117 estudios tras una revisión inicial. Posteriormente, en la tercera fase, de elegibilidad, se realizó una lectura detallada de los títulos, resúmenes y palabras clave, lo que llevó a una nueva depuración de los registros, quedando un total de 22 estudios, 13 de Scopus y 9 de WoS. En la cuarta fase, de inclusión, se consolidó la muestra definitiva de 19 estudios para la revisión, luego de eliminar 2 duplicados identificados en la fase anterior y excluir un trabajo que, si bien cumplía con los criterios establecidos, no se alineaba a los objetivos de esta investigación ya que no presentaba un vínculo claro entre Python y la educación matemática. Por último, esta muestra fue examinada en

profundidad mediante el sistema de categorías previamente descrito, obteniéndose así los datos base para el análisis cualitativo que se presenta a continuación.

### **Uso de Python en la Enseñanza y Aprendizaje de las Matemáticas**

Al revisar los estudios, se identificó que Python se ha incorporado en la enseñanza y aprendizaje de las matemáticas con diferentes propósitos, como en el diseño y desarrollo de cursos o la construcción de recursos y herramientas digitales. También, se destaca su uso en la aplicación de procesos matemáticos, entre ellos la visualización de conceptos, el modelado de situaciones, el desarrollo de proyectos interdisciplinarios y la resolución de problemas, siendo este último el de mayor relevancia según los datos encontrados.

Esto se evidencia en el estudio de Auccahuasi et al. (2018), ya que, a través de la programación, se proponen soluciones a problemas aritméticos como la suma de los primeros  $n$  números naturales. De tal manera, este estudio permitió explorar algunas funciones de Python para visualizar los resultados, lo que favoreció en los estudiantes la comprensión tanto de los procesos algorítmicos de programación como de los matemáticos. En esta misma línea, autores como Pulido-Rojano et al. (2023) muestran el uso de Python como una herramienta para abordar problemas de mayor complejidad, incluyendo modelos de la teoría de juegos, cadenas de Markov y teoría de colas, los cuales, si son resueltos únicamente con lápiz y papel, requerirían un tratamiento extenso y demandante (Pulido-Rojano et al., 2023), lo cual es contrario al incluir la programación en su proceso.

Desde un punto vista aplicado, se observó que el uso de Python se integra en la resolución de problemas reales y en el análisis de situaciones. Cirneanu y Moldoveanu (2024) proponen diversos escenarios, como el reconocimiento facial en contextos universitarios de seguridad de datos, donde este lenguaje de programación funciona como un entorno digital para implementar modelos matemáticos, permitiendo ajustarlos y entrenarlos con información para lograr una mayor presión en la resolución de dichas situaciones.

Asimismo, apuestas como la de Born (2015) revelan que Python se ha convertido en un recurso central en el diseño y desarrollo de cursos. En este caso, se implementa un laboratorio de matemáticas a nivel universitario, en el que los estudiantes trabajan en sus propios proyectos sobre procesamiento de señales de audio, control robótico, procesamiento de imágenes, simulaciones biológicas y geometría computacional. Esto refleja una amplia conexión entre la teoría matemática y las aplicaciones prácticas que se le dan a Python.

Por otra parte, se encontró la incorporación de Python en programas extracurriculares, particularmente, orientados a estudiantes de secundaria. El proyecto titulado *Avanzando el Aprendizaje Extraescolar en Matemáticas e Ingeniería* ejemplifica esta integración, en la que Python se apoya en dispositivos como Raspberry Pi para favorecer el aprendizaje de las matemáticas a través de representación de imágenes y videos. Estos se suelen mostrar como funciones discretas, introduciendo así nociones algebraicas, sistemas de coordenadas y representaciones numéricas (Pattichis et al., 2017).

Este tipo de experiencias no solo destacan por la comprensión conceptual que genera en los estudiantes, sino también, por la motivación que despiertan trabajar en entornos visuales e interactivos que involucran el uso de Python (Pattichis et al., 2017). Por otra parte, se identificó en el trabajo de Lvov et al. (2021) la incorporación de Python en el desarrollo de libros de textos electrónicos y aplicativos web con algoritmos para la resolución de problemas, incluso relacionado a otros lenguajes como C o Pascal. Esto ayudó a los estudiantes a interactuar directamente con los

contenidos matemáticos desde dispositivos y navegadores web, y no solo de manera manual (Lvov et al., 2021).

Al igual que Pulido-Rojano et al. (2023), el estudio de Inguva et al. (2021) evidencia un uso avanzado de Python, especialmente en el estudio de análisis numérico y en la resolución de ecuaciones diferenciales parciales mediante bibliotecas como FliPy. En este caso se busca reducir la barrera de entrada al aprendizaje de métodos numéricos complejos, ofreciendo códigos abiertos de Python que permiten a los estudiantes experimentar y visualizar diferentes soluciones, fortaleciendo así su comprensión en fundamentos del cálculo y del modelado matemático.

Asimismo, De Andrade y Fachada (2021) reportan experiencias en asignaturas de matemáticas discretas, donde se propone utilizar un motor tridimensional llamado *PyXYZ*, el cual es programado en Python, con el fin de observar y aplicar transformaciones 3D tales como: rotaciones, proyecciones, escalados y manipulación de matrices. Este tipo de propuesta facilitó implementar un motor funcional, promoviendo el uso práctico de la programación y de conceptos abstractos del álgebra lineal.

Desde una perspectiva transversal, se identificó que Python no actúa de manera aislada, sino en conjunto con plataformas digitales y con sus amplias librerías, como Numpy, SymPy, Pandas o Matplotlib, las cuales permiten realizar tanto visualizaciones como manipulaciones algebraicas y simulaciones (Gayoso et al., 2021; Lee et al., 2024; Ochkov et al., 2022). Todo ello evidencia un conjunto de finalidades pedagógicas en su vínculo con la educación matemática. Por su parte, propuestas como las de Koupritzioti y Xinogalos (2020) y de Shokaliuk y Kavetskyi (2025) muestran que la programación funciona como un recurso para apoyar el dominio de conceptos básicos en matemáticas, donde la resolución de problemas y el acceso a entornos digitales fortalecen la enseñanza y el aprendizaje de esta disciplina.

Así, Shokaliuk y Kavetskyi (2025) resaltan el uso de un tutor digital que ofrece retroalimentación inmediata a los estudiantes sobre sus conocimientos y promueve el aprendizaje de las matemáticas. Esta herramienta utiliza Python y CustomTKinter para generar preguntas y evaluar a los estudiantes de manera personalizada. Por otro lado, propuestas como la ya mencionada de Koupritzioti y Xinogalos (2020), fomentan entornos lúdicos y la integración de proyectos interactivos, donde los estudiantes desarrollan videojuegos como *PyDiophantus*, el cual se basa en un laberinto que combina operaciones aritméticas con la práctica de la programación.

Otra propuesta a nivel universitario es la de Liu y Castellana (2021), que, en la misma línea del trabajo de Born (2015), muestra el diseño y desarrollo de un curso estructurado alrededor de Python. Estos autores utilizan una extensión de Python llamada *DA* –una abreviación de lenguaje potente–, la cual permite trabajar con lógica proposicional, cuantificadores, conjuntos, lógica de predicado, entre otros temas. Esto ayudó a posicionar la programación como una herramienta que también sirve para verificar propiedades matemáticas.

Desde otra perspectiva, se ha encontrado que, en el trabajo Manne (2022), se diseñan marcos interdisciplinarios en los que Python se articula con la ingeniería de software para el modelado de fenómenos físicos y matemáticos. Estos modelos requieren un alto nivel de abstracción, lo que resalta el papel aplicado de Python para facilitar este proceso y favorecer el desarrollo de competencias profesionales en los estudiantes. Asimismo, el estudio de Abu-El-Haija et al. (2023) evidencia la evaluación de los aprendizajes en el contexto de un curso de ingeniería industrial, incorporando Python junto con otros recursos computacionales para medir el impacto de su uso en la comprensión de contenidos matemáticos e informáticos, lo que contribuyó a mejorar el rendimiento académico de los estudiantes.

A su vez, se hallaron investigaciones que resaltan el valor de utilizar Python para establecer conexiones curriculares en cursos académicos, vinculando las matemáticas con otras áreas del conocimiento. Así, por ejemplo, el estudio de Georgieva-Trifonova (2023) exhibe estas ideas, al emplear Python en la asignatura de *Modelado Computacional e Informática*, desarrollada en grado sexto de secundaria, para realizar proyectos interdisciplinarios. Los ejemplos trabajados reflejan tareas reales que incluyen el cálculo de perímetros y áreas, longitudes de circunferencias, descomposición de números, representación de sistemas de coordenadas, gráficos, entre otras relacionadas con geografía e historia.

Por último, se considera aquí la propuesta de Ali et al. (2025), la cual se destaca por la construcción de una herramienta digital inclusiva, programada en Python, destinada a la enseñanza y el aprendizaje de las matemáticas. Esta herramienta es un solucionador interactivo de expresiones diseñada para apoyar a estudiantes con discapacidad visual, promoviendo la independencia en la resolución de problemas matemáticos y facilitando un aprendizaje autónomo y accesible.

Cabe resaltar que, en términos educativos, estos hallazgos muestran que la mayoría de los estudios analizados se desarrollan a nivel universitario, donde Python se integra en carreras como ingeniería, matemáticas y ciencias aplicadas, favoreciendo el diseño y desarrollo de cursos, resolución de problemas complejos y la elaboración de proyectos interdisciplinarios. En contraste, se observa que son pocas las propuestas orientadas a educación secundaria, e incluso no se identificaron estudios que implementen este lenguaje de programación en etapas iniciales, a pesar de su potencial.

## **Tendencias y Enfoques Metodológicos**

El análisis también permitió identificar las principales tendencias y enfoques metodológicos asociados al uso de Python en la enseñanza y aprendizaje de las matemáticas. La evidencia muestra un predominio de enfoques innovadores, como el aprendizaje basado en proyectos (ABP), el aprendizaje activo y las apuestas de la Educación STEAM (lo que en español es conocido como Ciencias, Tecnología, Ingeniería, Arte y Matemáticas). Además, se identificaron otros enfoques como la gamificación, el aprendizaje experimental y práctico, así como propuestas emergentes, incluyendo el aprendizaje inclusivo y la aplicación de principios de la neuroeducación.

En este sentido, estudios como el de Auccahuasi et al. (2018) adoptan fundamentos de la neuroeducación para diseñar estrategias que consideran el funcionamiento del cerebro con el fin de motivar el interés de los estudiantes en matemáticas e informática, mostrando así cómo problemas matemáticos pueden abordarse con lógica de programación. Otra propuesta, implementa entornos gamificados en los que no solo se integra un tutor personalizado para la interacción con los estudiantes, sino que también, se incorporan elementos como recompensas o barras de progresos, los cuales fomentan en los estudiantes la participación y la motivación en la resolución de problemas (Shokaliuk y Kavetskyi, 2025).

Asimismo, se identificaron propuestas que vinculan los principios STEM en la construcción de experiencias de aprendizaje basadas en proyectos. Un ejemplo es la apuesta curricular de Born (2015), desde la cual se busca desarrollar capacidades de modelación y resolución de mediante proyectos en los que los estudiantes seleccionan herramientas, trabajan en equipos y experimentan por medio ensayos de prueba y error en Python. También, desde una perspectiva interdisciplinar, el trabajo de Pattichis et al. (2017) muestra experiencias donde los

estudiantes integran las matemáticas con el lenguaje de programación para generar representación de imágenes y videos, evidenciando así un enfoque de trabajo colaborativo y aplicado.

Estos principios se amplían en apuestas que combinan la educación STEM y STEAM, al integrar el arte y la creatividad mediante visualizaciones matemáticas, donde la resolución de problemas se articula con simulaciones y aprendizajes activos en los estudiantes (Ochkov et al., 2022). Además, se identificaron enfoques de aprendizaje aplicado y experimental que, aunque no se denominan explícitamente como metodología del ABP, se alinea con sus principios y favorecen la construcción de simulaciones, el trabajo en proyectos auténticos y la producción de código funcional en Python para resolver problemas prácticos. Esto también se evidencia en experiencias de aprendizaje activo en matemáticas discretas y en la resolución de problemas en ingeniería y desarrollo de simulaciones interdisciplinarias (De Andrade y Fachada, 2021; Gayoso et al., 2021; Liu y Castellana, 2021; Manne, 2022).

Además, se identificaron apuestas educativas inclusivas que integran proyectos auténticos y de aprendizaje colaborativos, donde Python se utiliza para garantizar la interacción y accesibilidad de las matemáticas, favoreciendo entornos de aprendizajes equitativos (Lee et al., 2024). En esta misma línea, Ali et al. (2025) muestran la creación de herramientas destinadas a apoyar a estudiantes con discapacidad visual en el aprendizaje de las matemáticas, lo cual evidencia la emergencia de estos enfoques orientados a la inclusión educativa.

Tanto en las intervenciones pedagógicas como en la planificación de cursos se identificaron algunos cursos estructurados alrededor de proyectos, con fases iniciales de exploración y etapas posteriores de desarrollo. En estos, los estudiantes seleccionan sus proyectos y presentan un producto final, lo que promueve un aprendizaje activo y basado en la investigación (Born, 2015). En el ámbito extraescolar, se encuentran propuestas innovadoras como la de López Leiva et al. (2022), la cual incorpora un modelo de aprendizaje práctico y colaborativo, donde los estudiantes asumen el rol de tutores para fortalecer tanto los conocimientos en matemáticas como en programación en contextos culturalmente sostenibles.

Por último, se observa una limitada presencia de metodologías tradicionales, ya que funciona como contraste frente a los enfoques innovadores previamente mencionados. Aunque se identificaron estudios que conservan métodos convencionales, como clases teóricas o memorización de contenidos, el uso de recursos tecnológicos, como libros electrónicos o prácticas complementarias, permite que Python se integre como herramienta para promover un aprendizaje activo, práctico y dinámico, rompiendo con la metodología tradicional (Abu-El-Haija et al., 2023; Cirneanu y Moldoveanu, 2024; Lvov et al., 2021; Manne, 2022). Asimismo, Liu y Castellana (2021) resaltan que este lenguaje de programación permite superar limitaciones de cálculo manual y de memorización mecánica, acercando la matemática a la práctica mediante código.

### **Beneficios y Limitaciones del Vínculo entre Python y las Matemáticas**

El análisis realizado permitió reconocer que el uso de Python en la enseñanza y el aprendizaje de las matemáticas aporta beneficios que se extienden tanto a la comprensión conceptual y desarrollo de habilidades como a aspectos motivacionales y afectivos en los estudiantes. En los estudios abordados se evidencia que Python facilita el tránsito de lo abstracto a lo concreto al posibilitar la exploración de funciones y visualizaciones de los resultados matemáticos.

De esta manera, los estudiantes pueden comprender de forma más clara los procesos matemáticos que, de otro modo, resultarían demasiado teóricos o distantes de su experiencia cotidiana, ya que la programación les permite desarrollar en conjunto habilidades matemáticas y

de lógica de programación (Auccahuasi et al., 2018). En esta misma línea, se encontró que Python permite abordar problemas matemáticos complejos y, al mismo tiempo, desarrollar competencias digitales vinculadas al trabajo con simulaciones y modelado de situaciones (Pulido-Rojano et al., 2023). Estos estudios revelan que la integración de Python no solo fortalece el dominio matemático, sino que también, favorece las conexiones interdisciplinarias para la formación actual (Cirneanu y Moldoveanu, 2024).

Otros hallazgos apuntan a la capacidad que tiene Python para promover aprendizajes autónomos. La interacción con entornos digitales y la posibilidad de recibir retroalimentación inmediata favorece la exploración y el descubrimiento, permitiendo que los estudiantes verifiquen sus hipótesis en tiempo real (Shokaliuk y Kavetskyi, 2025). Además, se observó que, en diferentes contextos universitarios, esta dinámica genera una comprensión más profunda de los contenidos y fomenta la disposición de los estudiantes hacia el aprendizaje independiente y a su propio ritmo (Gayoso et al., 2021).

Asimismo, la inclusión de proyectos prácticos en los que los estudiantes vinculan las matemáticas con situaciones cercanas a su realidad, potencia la motivación y refuerza la percepción de utilidad del conocimiento matemático; además, se fomenta el liderazgo y el rendimiento académico (López Leiva et al., 2022). De este modo, los beneficios también se evidencian en la asimilación de los contenidos matemáticos complejos que tradicionalmente resultan de difícil comprensión para los estudiantes.

Experiencias en cursos de análisis numérico, álgebra lineal y matemáticas discretas muestran que Python, mediante la simulación y visualización, ayuda a reducir la abstracción excesiva de ciertos conceptos y facilita el tránsito al aprendizaje más aplicado (De Andrade y Fachada, 2021; Inguva et al., 2021). Este mismo potencial se observó en la incorporación de ecuaciones diferenciales, donde el uso de bibliotecas de Python permitió experimentar con diferentes métodos de solución para comprender sus alcances y limitaciones (Inguva et al., 2021).

Otro beneficio identificado es la inclusión educativa, pues el diseño de herramientas digitales programadas en Python ha favorecido la participación de estudiantes con discapacidad visual en la resolución de problemas matemáticos, garantizando así la autonomía y accesibilidad en el proceso de aprendizaje (Ali et al., 2025). De igual manera, el trabajo colaborativo en torno a proyectos programados en Python fortalece la construcción de aprendizaje significativo y la confianza en las capacidades propias de los estudiantes (Ali et al., 2025).

Esto demuestra que Python, en lo que refiere a la integración de proyectos, favorece los vínculos interdisciplinarios, contribuyendo no solo al fortalecimiento de competencias matemáticas, sino también, a la creatividad y motivación, al vincular dicha disciplina con contextos significativos para los estudiantes (Georgieva-Trifonova, 2023). En programas extracurriculares, la integración de las matemáticas con la programación contribuyó a la construcción de identidades positivas en STEM, especialmente en estudiantes de grupos subrepresentados (Gayoso et al., 2021).

Ahora bien, a pesar de estos beneficios mencionados y del gran potencial de Python, se identificaron algunas limitaciones. De esta manera, se encontró que los estudiantes enfrentan dificultades iniciales relacionadas con la sintaxis del lenguaje y la construcción de expresiones, lo que requiere apoyos complementarios para garantizar el aprendizaje fluido de los contenidos (Born, 2015). En el contexto universitario, las desigualdades en la preparación previa en programación han generado brechas en el desempeño de los estudiantes, dificultando la participación equitativa y demandando un mayor acompañamiento docente (De Andrade y Fachada, 2021). Asimismo, se reconocieron limitaciones contextuales vinculadas a la falta de

recursos técnicos adecuados, como conectividad estable y dispositivos de alto rendimiento, que condicionan la implementación de estas propuestas en algunos contextos (Lvov et al., 2021).

Otro desafío relevante es la necesidad de formación docente. La literatura señalada advierte que, si bien Python representa una oportunidad para enriquecer la enseñanza de las matemáticas, la falta de preparación suficiente por parte de los educadores puede dificultar su integración curricular (Cirneanu y Moldoveanu, 2024). En este sentido, también se identificó la falta de lineamientos institucionales claros que respalden de manera sistemática la inclusión de la programación en el área de las matemáticas, lo cual limita la sostenibilidad de propuestas educativas (Lvov et al., 2021).

Estos hallazgos permiten afirmar que el vínculo entre Python y las matemáticas aporta beneficios significativos en términos de comprensión conceptual, motivación, autonomía, interdisciplinariedad e inclusión. No obstante, estos resultados también ponen en evidencia limitaciones relacionadas con factores técnicos, contextuales e institucionales, que deben ser abordados para garantizar una implementación más equitativa y sostenible en distintos niveles educativos.

## Discusión

Los hallazgos en esta revisión sistemática de la literatura permiten afirmar que la integración de Python en la enseñanza y el aprendizaje de las matemáticas se enmarca en una tradición pedagógica. A partir de Papert (1980) se ha mostrado la necesidad de superar los enfoques tradicionales dominantes y pasar de la memorización mecánica a promover un aprendizaje activo y significativo en los estudiantes. La evidencia recogida muestra que Python se utiliza para favorecer la resolución de problemas, la experimentación y el vínculo interdisciplinar en el diseño de cursos y proyectos.

Esto confirma la vigencia de la propuesta de Papert en torno a la programación como medio para pensar matemáticamente y motivar a los estudiantes, y no solo como un fin instrumental. La motivación y la creatividad reportadas en los estudios mencionados se alinean con la visión de la matemática como un espacio para la exploración y el descubrimiento, y no como una disciplina abstracta y distante de los contextos de los estudiantes.

Al contrastar los resultados con los fundamentos teóricos previamente expuestos, se observa que el uso de Python materializa de manera concreta el vínculo entre el PC y el pensamiento matemático señalado por diversos autores. Por ejemplo, Wing (2006, 2008) plantea que el PC implica la descomposición de problemas complejos en elementos más manejables para la formulación de soluciones. Los estudios revisados evidencian también cómo mediante el uso de las diferentes bibliotecas de Python, los estudiantes pueden modelar, simular y visualizar situaciones matemáticas, lo que se traduce en la práctica de las acciones de abstracción y generalización descritas por Wing (2006, 2008). De esta manera, se muestra que Python no se limita a ser un lenguaje accesible por su sintaxis, sino que se convierte en un medio que potencia habilidades cognitivas centrales tanto en el pensamiento matemático como en el computacional.

De igual forma, los resultados hallados coinciden con lo señalado por Auccahuasi et al. (2018) y Betteridge et al., (2022), quienes sostienen que programar constituye una vía eficiente para aprender matemáticas, sin dejar de lado que ello también requiere conocimientos en la disciplina. En esta misma línea, experiencias como las de Born (2015) o Inguava et al. (2021), que emplean Python para abordar contenidos de cálculo, álgebra lineal o análisis numérico, confirman que la programación permite reducir la distancia entre la teoría y la aplicación práctica, lo cual facilita el tránsito de lo abstracto a lo concreto en nociones tradicionalmente complejas,

permitiendo a los estudiantes alcanzar una mejor comprensión de las matemáticas. No obstante, ante esto emergen tensiones que dialogan con las barreras ya anticipadas, por ejemplo, Kaufmann et al. (2023) señalan la falta formación docente y la resistencia a modificar las metodologías tradicionales que limitan la integración de la programación en la enseñanza y aprendizaje de las matemáticas.

En este caso, los estudios revisados permiten ver que Python ofrece un gran potencial pedagógico en la didáctica de las matemáticas; sin embargo, su implementación suele concentrarse en la enseñanza universitaria y depende de condiciones contextuales, como la disponibilidad de recursos tecnológicos o el acompañamiento docente especializado (Kaufmann et al., 2023). Estas limitaciones sugieren que, para lograr que Python transforme la enseñanza y aprendizaje de las matemáticas, se requiere de políticas educativas y programas de formación docente que respalden su uso sostenido y equitativo en diferentes contextos.

Asimismo, esta revisión de literatura muestra aportes novedosos en relación con la motivación y la inclusión educativa. Autores como Betteridge et al. (2022) subrayan la capacidad de Python para motivar a los estudiantes y permitirles encontrar un sentido a las matemáticas. Estas ideas se confirman en apuestas que vinculan la programación en entornos lúdicos, gamificados o interdisciplinarios, en los que el aprendizaje de las matemáticas deja de ser un ejercicio aislado y adquiere relevancia en contextos significativos.

Por último, estos hallazgos no solo refuerzan fundamentos teóricos, sino que también, actúan como puente entre el pensamiento matemático y computacional, tal como lo señalan De la Hoz e Hijón (2022), Resnick et al., (2009) y Rivera Muñoz (2021), al mostrar que la programación fomenta la experimentación, el razonamiento lógico y el trabajo colaborativo. Sin embargo, la escasa presencia de este lenguaje en niveles escolares y las limitaciones identificadas permiten ver que aún es necesario fortalecer su integración curricular para que su impacto sea efectivo y pueda atender las barreras y demandas actuales de los estudiantes.

## **Conclusiones**

Esta investigación tuvo como propósito examinar el papel que juega el lenguaje de programación Python en la enseñanza y el aprendizaje de las matemáticas, mediante una revisión sistemática de la literatura, desarrollada con base en el protocolo PRISMA. El análisis de los estudios permitió reunir evidencia empírica y teórica que aporta una visión amplia sobre sus diferentes usos, alcances y limitaciones en diversos contextos educativos. Esto permitió reconocer el vínculo inherente entre la programación y las matemáticas, pero también, puso de manifiesto limitaciones que deben atenderse para favorecer su incorporación en las aulas.

Aunque la revisión de literatura enfrentó dificultades, como la adquisición de ciertos estudios que no eran de acceso abierto y las complicaciones para abordar lecturas en inglés, estas no representaron una barrera significativa. Dichos obstáculos se superaron mediante estrategias complementarias, como la comunicación con colegas para obtener los estudios de acceso restringido y el uso de herramientas de traducción que facilitaron la comprensión de los trabajos, asegurando así un corpus sólido para el análisis.

Este proceso de revisión permitió reconocer que el uso de Python en la educación matemática se vincula, en particular, con las metodologías activas y los enfoques innovadores como el ABP, la gamificación y la Educación STEAM, así como con enfoques emergentes, entre ellos el inclusivo. Esto ayudó a los estudiantes a comprender conceptos, aplicar las matemáticas en situaciones reales y desarrollar su motivación al trabajar en proyectos auténticos en escenarios de experimentación y resolución de problemas, favoreciendo así un aprendizaje activo y

significativo. Esto es coherente con las ideas que se han desarrollado a partir de Papert (1980), sobre la necesidad de superar la memorización y promover experiencias significativas en los estudiantes.

Asimismo, se evidenció que Python contribuye a la comprensión de los contenidos abstractos al permitir modelar, simular y visualizar procesos matemáticos. También, se encontró que la integración de Python en las matemáticas aumenta positivamente la motivación, la creatividad y la inclusión en los estudiantes, como lo evidencian experiencias que incorporan entornos lúdicos o accesibles para quienes presentan condiciones contextuales particulares, como los grupos subrepresentados (Pattichis et al., 2017) o aquellos con discapacidades, entre ellas la visual (Ali et al., 2025). Sin embargo, persisten obstáculos asociados a la falta de formación docente, las brechas en preparación del estudiantado y el difícil acceso a recursos tecnológicos. Estos hallazgos ponen en evidencia que, aunque el potencial pedagógico de Python es notorio, su implementación seguirá siendo limitada mientras no existan políticas educativas, programas de formación docente y una inversión sostenida en recursos tecnológicos, de forma que llegue a ser accesible para todos (Kaufmann et al., 2023).

Los resultados también evidenciaron que Python potencia diversas habilidades digitales y matemáticas, entre ellas el pensamiento estructurado y algorítmico, la abstracción, la modelación, la visualización y el razonamiento lógico. Esto facilita el desarrollo del pensamiento matemático y computacional, en coherencia con lo planteado por Wing (2006, 2008). Así, más allá de reforzar contenidos específicos, el uso de este lenguaje de programación promueve una forma de pensar estructurada y crítica, lo que lo convierte en una herramienta pedagógica para vincular el pensamiento matemático y el PC, y de fortalecer competencias clave para la educación actual. En particular, este vínculo ha tomado mayor fuerza en la educación universitaria, respondiendo a las demandas de los estudiantes y permitiendo el diseño y desarrollo de cursos en los que se abordan situaciones y se resuelven problemas complejos, lo que contribuye al desarrollo de habilidades y competencias en los estudiantes. Además, se llevan a cabo proyectos que integran diferentes disciplinas, lo que permite una aplicación amplia y un vínculo más significativo entre la programación y las matemáticas. No obstante, su presencia en otros contextos educativos aún es limitada, lo que puede poner una barrera en el panorama.

Dicho lo anterior, la revisión de literatura evidencia diferentes elementos para responder a la pregunta central, al mostrar que Python se ha usado como un recurso didáctico eficaz para mejorar la comprensión de las matemáticas y fomentar el pensamiento computacional. En este sentido, su uso en las aulas promueve la motivación de los estudiantes para aprender no solo programación, sino también matemáticas. En términos de aportes, esta investigación ofrece un panorama actualizado y sistemático sobre la evidencia disponible en la literatura, lo cual resulta útil tanto para futuras investigaciones como para docentes interesados en explorar y vincular la programación a la didáctica de las matemáticas. No obstante, permanece abierta la necesidad de continuar explorando su aplicación en niveles educativos como primaria y secundaria, e incluso en la formación del profesorado; así como de diseñar investigaciones que midan con mayor precisión el impacto de Python en el rendimiento académico, la motivación y el desarrollo de competencias digitales de los estudiantes.

Finalmente, a modo de recomendación, se sugiere a los docentes y responsables de políticas educativas considerar la programación no solo como un contenido adicional, sino como un recurso estratégico para potenciar el aprendizaje y la enseñanza de las matemáticas. Esto favorece en los estudiantes el desarrollo de habilidades para responder a las necesidades educativas y sociales. Para futuras investigaciones, se sugiere diversificar los contextos de estudio e incluir metodologías

y enfoques mixtos que integren no solo percepciones y experiencias, sino también, resultados cuantitativos sobre su impacto en las aulas.

### Referencias

- Abu-El-Haija, L., Qaadan, S. y Ammouri, M. (octubre, 2023). *Effect Of Computer Aided Math In Industrial Engineering Curricula, A Case Study*. Trabajo presentado en Conference: CIE50, merican University of Sharjah, Emiratos Árabes Unidos.
- Ali, A., Khusro, S., Alam, F., Khan, I. y Algamdi, S. A. (2025). AIME-solve: Accessible interactive mathematical expressions solver for optimized learning and visual empowerment. *SoftwareX*, 30, 102143. <https://doi.org/10.1016/j.softx.2025.102143>
- Auccahuasi, W., Bernardo, G., Bernardo, M., Vega, D., Urbano, R. y Oré, E. (Julio, 2018). *Herramienta interactiva en línea como instrumento para el aprendizaje de las matemáticas mediante las técnicas de programación, dirigido a alumnos de educación secundaria*. Trabajo presentado en 16th LACCEI International Multi-Conference for Engineering, Education, and Technology: Innovation in Education and Inclusion, Lima, Perú. <http://dx.doi.org/10.18687/LACCEI2018.1.1.573>
- Auccahuasi, W., Santiago, G. B., Núñez, E. O. y Sernaque, F. (2018). *Interactive online tool as an instrument for learning mathematics through programming techniques, aimed at high school students*. Trabajo presentado en 6th International conference on information technology: IoT and Smart City (pp. 70-76). <https://doi.org/10.1145/3301551.3301580>
- Betteridge, J., Chan, E., Corless, R. M., Davenport, J. H. y Grant, J. (2022). Teaching programming for mathematical scientists. En P. R. Richard, M. P. Veléz, y S. Vaerenbergh (Eds.) *Mathematics Education in the Age of Artificial Intelligence: How Artificial Intelligence can Serve Mathematical Human Learning* (pp. 251-276). Springer Cham. [https://doi.org/10.1007/978-3-030-86909-0\\_12](https://doi.org/10.1007/978-3-030-86909-0_12)
- Born, S. (2015). *A mathematical lab for first year students*. Trabajo presentado en 43rd Annual SEFI Conference, Orleans, Francia.
- Bosch Saldaña, M. A. (2021). Apuntes teóricos sobre el pensamiento matemático y multiplicativo en los primeros niveles. *Edma 0-6: Educación Matemática en la infancia*, 1(1), 15-37. <https://doi.org/10.24197/edmain.1.2012.15-37>
- Cirneanu, A.-L. y Moldoveanu, C.-E (2024). Use of Digital Technology in Integrated Mathematics Education. *Applied System Innovation*, 7(4). <https://doi.org/10.3390/asi7040066>
- Creswell, J. W. (2013). *Qualitative inquiry and research design: Choosing among five approaches*. Sage publications.
- De Andrade, D. y Fachada, N. (2021). *PyXYZ: an educational 3D wireframe engine in Python*. Trabajo presentado en 26th ACM Conference on Innovation and Technology in Computer Science Education V.1 (pp. 519-525). <https://doi.org/10.1145/3430665.3456345>
- De la Hoz Ruiz, A. y Hijón Neira, R. (2022). Enseñanza de las Matemáticas a través del uso de Scratch (Transversalidad STEM). *IE Comunicaciones: Revista Iberoamericana de Informática Educativa*, 36, 14-34. <https://dialnet.unirioja.es/servlet/articulo?codigo=8732468>

- Ferreira González, I., Urrútia, G. y Alonso-Coello, P. (2011). Systematic reviews and meta-analysis: scientific rationale and interpretation. *Revista Española de Cardiología (English Edition)*, 64(8), 688-696. <https://doi.org/10.1016/j.recesp.2011.03.029>
- Gayoso Martínez, V., Hernández Encinas, L., Martín Muñoz, A., y Queiruga Dios, A. (2021). Using free mathematical software in engineering classes. *Axioms*, 10(4), 253. <https://doi.org/10.3390/axioms10040253>
- Georgieva-Trifonova, T. (2023). Establishing Cross-Curricular Connections Through Python Programming in Computer Modeling Education. *TEM Journal*, 12(3), 1786-1791. <https://doi.org/10.18421/TEM123-61>
- Gillespie, C. W. (2004). La visión de Seymour Papert para la educación de la niñez: Estudio descriptivo de estudiantes de Head Start y kindergarten en aulas basadas en el descubrimiento y enriquecidas con el lenguaje de programación Logo. *ECRP: Investigación y Práctica de la Niñez Temprana*, 6(1). <https://ecrp.illinois.edu/v6n1/gillespie-sp.html>
- González Arias, C. (2011). La formulación de los objetivos en artículos de investigación científica en cuatro disciplinas: historia, lingüística, literatura y biología. *Linguagem em (Dis) curso*, 11(2), 401-429. <https://doi.org/10.1590/S1518-76322011000200010>
- Grover, S. y Pea, G. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, 42 (1), 38-43. <https://doi.org/10.3102/0013189X12463051>
- Inguva, P., Bhute, V. J., Cheng, T. N. y Walker, P. J. (2021). Introducing students to research codes: A short course on solving partial differential equations in Python. *Education for Chemical Engineers*, 36, 1-11. <https://doi.org/10.1016/j.ece.2021.01.011>
- Kaufmann, O. T., Maugesten, M. y Meaney, T. (2023). The professional identities of prospective mathematics teachers who teach through programming. *Journal of Mathematics Teacher Education*, 27(6), 1039-1053. <https://doi.org/10.1007/s10857-023-09594-3>
- Koupritzioti, D. y Xinogalos, S. (2020). PyDiophantus maze game: Play it to learn mathematics or implement it to learn game programming in Python. *Education and Information Technologies*, 25(4), 2747-2764. <https://doi.org/10.1007/s10639-019-10087-1>
- Lee, W., Moh, M., Khayrallah, R., Attar, N. y Lam, K. (2024). *WIP: Python for Everyone as a Mathematics GE Course: Broaden Participation and Enhance Data Science Career Pipeline*. Trabajo presentado en 2024 IEEE Frontiers in Education Conference (FIE) (pp. 1-5). IEEE. <https://doi.org/10.1109/FIE61694.2024.10892893>
- Liu, Y. A. y Castellana, M. (2021). *Discrete math with programming: A principled approach*. Trabajo presentado en 52nd ACM Technical Symposium on Computer Science Education (pp. 1156-1162). <https://doi.org/10.48550/arXiv.2011.14059>
- López Leiva, C. A., Noriega, G., Celedón-Pattichis, S., y Pattichis, M. S. (2022). From students to cofacilitators: Latinx students' experiences in mathematics and computer programming. *Teachers College Record*, 124(5), 146-165. <https://doi.org/10.1177/01614681221104104>
- Lvov, M., Kravtsov, H., Shishko, L., Hniedkova, O., Chernenko, I. y Kozlovsky, E. (septiembre, 2021). About Electronic Textbook “Mathematical Tasks Programming. First Steps”. En G. Li, J. Filipe y Z- Xu (Eds.) *Communications in Computer and Information Science*

- (pp. 334-349). Cham: Springer International Publishing. [https://doi.org/10.1007/978-3-031-14841-5\\_22](https://doi.org/10.1007/978-3-031-14841-5_22)
- Manne, H. (2022). Applied Perspectives of Physical and Mathematical Systems with Ordinary Software Engineering Tools. Publicado en 2022 *12th International Conference on Pattern Recognition Systems (ICPRS)* (pp. 1-6). IEEE. <https://doi.org/10.1109/ICPRS54038.2022.9854061>
- Marín, V. I. (2022). La revisión sistemática en la investigación en Tecnología Educativa: observaciones y consejos. *RiiTE Revista interuniversitaria de investigación en Tecnología Educativa*, 13, 62-79. <https://doi.org/10.6018/riite.533231>
- Ochkov, V. F., Stevens, A. y Tikhonov, A. I. (2022). *Jupyter Notebook, JupyterLab–Integrated Environment for STEM Education*. Trabajo presentado en 2022 VI International Conference on Information Technologies in Engineering Education (Inforino) (pp. 1-5). <https://doi.org/10.1109/Inforino53888.2022.9782924>
- Palma Suárez, C. A. y Sarmiento Porras, R. E. (2015). Estado del arte sobre experiencias de enseñanza de programación a niños y jóvenes para el mejoramiento de las competencias matemáticas en primaria. *Revista mexicana de investigación educativa*, 20(65), 607-641. [https://www.scielo.org.mx/scielo.php?pid=S1405-66662015000200013yscript=sci\\_abstract](https://www.scielo.org.mx/scielo.php?pid=S1405-66662015000200013yscript=sci_abstract)
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.
- Pattichis, M. S., Celedon-Pattichis, S. y LopezLeiva, C. (2017). *Teaching image and video processing using middle-school mathematics and the Raspberry Pi*. Trabajo presentado en 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6349-6353). IEEE. <https://doi.org/10.1109/ICASSP.2017.7953378>
- Pinargote-Zambrano, J. J., Lino-Calle, V. A. y Vera-Almeida, B. J. (2024). Python en la enseñanza de las Matemáticas para estudiantes de nivelación en Educación Superior. *MQRInvestigar*, 8(3), 3966–3989. <https://doi.org/10.56048/MQR20225.8.3.2024.3966-3989>
- Pulido-Rojano, A. D., Rodríguez-Herazo, J., Madera-Narvaez, O., Maldonado-Muñoz, D., Cano-Camacho, E., De-La-Hoz-Arroyo, J. y Henríquez-Gomez, M. (2023). Nueva Herramienta Computacional para la Enseñanza en la Solución de Problemas de Investigación de Operaciones. En N. Callaos, J. Horne, E. F. Ruiz-Ledesma, B. Sánchez, A. Tremante (Eds.) *Memorias de la Décima Tercera Conferencia Iberoamericana de Complejidad, Informática y Cibernética: CICIC 2023* (pp. 85-90). International Institute of Informatics and Cybernetics. <https://doi.org/10.54808/CICIC2023.01.85>
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., et al. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60-67. <https://doi.org/10.1145/1592761.1592779>
- Rivera Muñiz, B. (2021). *Los lenguajes de programación como medio para la enseñanza de las matemáticas*. Trabajo presentado en Seminario Interuniversitario de Investigación en Ciencias Matemáticas, Puerto Rico.
- Sánchez Serrano, S., Pedraza Navarro, I. y Donoso González, M. (2022). ¿Cómo hacer una revisión sistemática siguiendo el protocolo PRISMA?: Usos y estrategias fundamentales

- para su aplicación en el ámbito educativo a través de un caso práctico. *Bordón: Revista de pedagogía*, 74(3), 51-66. <https://doi.org/10.13042/Bordon.2022.95090>
- Shokaliuk, S. V. y Kavetskyi, A. O. (2025). *Design and evaluation of a personalized digital mathematics tutor for grade 6 learners*. Trabajo presentado en CEUR Workshop Proceedings (pp. 58-65). <https://elibrary.kdpu.edu.ua/xmlui/handle/123456789/12021>
- Tejera-Martínez, F., Aguilera, D. y Vílchez-González, J. M. (2020). Lenguajes de programación y desarrollo de competencias clave. Revisión sistemática. *Revista electrónica de investigación educativa*, 22. <https://doi.org/10.24320/redie.2020.22.e27.2869>
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725. <https://doi.org/10.1098/rsta.2008.0118>
- Wing, J. (2006). Pensamiento computacional. *Comunicaciones de la ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>