

65

Fecha de presentación: julio, 2021
Fecha de aceptación: septiembre, 2021
Fecha de publicación: octubre, 2021

ESTRATEGIAS DIDÁCTICAS

EN EL PROCESO DE ENSEÑANZA-APRENDIZAJE DE LA PROGRAMACIÓN

DIDACTIC STRATEGIES IN THE PROCESS OF TEACHING-LEARNING OF THE PROGRAMMING

Ángela Sarría Stuart¹

E-mail: asarria@ucf.edu.cu

ORCID: <https://orcid.org/0000-0001-8479-7392>

Ariel Gómez Sarría¹

E-mail: agomez@ucf.edu.cu

ORCID: <https://orcid.org/0000-0003-3207-084X>

Ailec Granda Dihigo²

E-mail: ailec@uci.cu

ORCID: <https://orcid.org/0000-0001-9009-5899>

¹ Universidad de Cienfuegos "Carlos Rafael Rodríguez" Cuba.

² Universidad de las Ciencias Informáticas. La Habana. Cuba.

Cita sugerida (APA, séptima edición)

Sarría Stuart, Ángela, Gómez Sarría, A., & Granda Dihigo, A. (2021). Estrategias didácticas en el proceso de enseñanza-aprendizaje de la programación. *Revista Universidad y Sociedad*, 13(S2), 549-556.

RESUMEN

En el contexto actual, la humanidad experimenta un gran desarrollo científico-tecnológico, en especial el área de las tecnologías de la información y las comunicaciones. En consecuencia, existe mayor necesidad de capital humano preparado en Informática. Como parte de dicha ciencia, la programación de computadoras es considerada, una habilidad fundamental para los nativos digitales, no solo los que realicen estudios profesionales. Varios autores refieren las dificultades que presentan los estudiantes para vencer dicho contenido. Ante esta situación, disímiles estudios se han efectuado para encontrar alternativas que hagan posible la obtención de mejores resultados en la conducción del proceso de enseñanza-aprendizaje en cuestión. En la comunicación, a partir de una investigación documental de fuentes relacionadas con la temática referida, se presenta una sistematización de estrategias didácticas empleadas para tales fines.

Palabras clave: Proceso de enseñanza-aprendizaje de la programación, estrategias didácticas, importancia de la programación de computadoras.

ABSTRACT

In the current context, the humanity experiences a great scientific-technological development, especially the area of the technologies of the information and the communications. In consequence, bigger necessity of prepared human capital exists in Computer science. As part of this science, the programming of computers is considered, a fundamental ability that the native ones digital they should possess, not alone those that carry out professional studies. Several authors refer the difficulties that the students present to conquer this content. Before this situation, dissimilar studies have been made to find alternative that make possible the obtaining of better results in the conduction of the teaching-learning process in question. In the communication, starting from a documental investigation carried out on referred bibliographical sources related with the thematic one, a systematizing of strategies didactic employees is presented for such ends.

Keywords: Process of teaching-learning of the programming, didactic strategies, importance of the programming of computers.

INTRODUCCIÓN

La aparición de la primera máquina computadora electrónica clasifica como uno de los logros más significativos del siglo XX. Fue un momento trascendental en la búsqueda de instrumentos que apoyaran al hombre en la realización de complejos cálculos. Este descubrimiento contribuyó al desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC).

En este contexto, el progreso que experimenta la humanidad, en la ciencia y la tecnología (en especial- dado el objetivo de este trabajo- las TIC) ofrecen oportunidades significativas para el desarrollo mundial. Específicamente en Cuba, la informatización de la sociedad, está en el centro de las políticas, tanto en el sector empresarial como en la administración pública.

Para cumplir tal propósito, las universidades tienen un rol fundamental a partir de los procesos sustantivos que en ellas se desarrollan. Por eso, en la estrategia que sigue el país, se presta atención a la formación de pregrado y de posgrado de ingenieros, licenciados o técnicos informáticos y de otras carreras afines que contribuirán al desarrollo de la industria del software. En Cuba son diversas las carreras de la Educación Superior encargadas de formar profesionales relacionados con la ciencia Informática, entre ellas: Ingeniería Informática, la Automática, Ciencias de la Computación, Telecomunicaciones y Licenciatura en Educación, especialidad Informática.

El análisis realizado por los autores al currículo de la formación de ingenieros o licenciados informáticos en el mundo, devela que la disciplina Programación es común. Por lo general, aparece desde los primeros semestres. La resolución independiente de problemas a partir de la búsqueda de un algoritmo, es el objetivo básico de la enseñanza de la programación. Su impacto es significativo en el campo profesional del egresado. Así lo expresan Fábrega, et al. (2016a), *“la era de la información se cristalizó en la vida cotidiana de las personas y aumentó la complejidad e inmediatez de sus interacciones con otros. Todos estos aumentos en escala e intensidad en el flujo de información han expuesto a la población a una avalancha de datos que es imposible de procesar completamente. Por ello, es natural que haya surgido una demanda por métodos que permitan transformar esos datos en significado y no mero ruido. Quienes saben programar han sido vitales para generar esos métodos”* (p.5)

Estudios efectuados por los autores evidencian que en la actualidad existe una mayor conciencia en torno a que introducir la enseñanza de la programación en las escuelas genera impactos cognitivos, facilita el desarrollo de habilidades para resolver tareas y promueve el pensamiento lógico. Entre otras razones, programar es considerada como una habilidad fundamental en el siglo XXI, no solo como un tema para profesionales de las Ciencias de la Computación.

Pese a las bondades señaladas, la *programación de computadoras* es una actividad que implica un complejo y creativo proceso mental. Exige inteligencia, conocimiento, habilidades y disciplina. Constituye una de las asignaturas más difíciles. Es esa la apreciación también

de Compañ, et al. (2015); González, et al. (2015); y otros investigadores.

Es frecuente que las asignaturas que integran las disciplinas relacionadas con la programación, tengan un alto grado de estudiantes con bajo aprovechamiento académico, esto es causa en no pocos casos, de la deserción estudiantil. Se justifica entonces que en la literatura especializada aparezcan variadas investigaciones sobre el proceso de enseñanza de la programación. El resultado que se presenta es una contribución parcial, de una investigación más amplia que se desarrolla en la Universidad de Cienfuegos sobre la enseñanza y el aprendizaje de la programación. El propósito de este artículo es ofrecer un acercamiento al proceso de enseñanza-aprendizaje de la programación, desde los resultados de autores que lo han estudiado. A partir de una investigación documental de fuentes relacionadas con la temática referida, se presentan como resultado una sistematización de estrategias didácticas empleadas para tales fines.

MATERIALES Y MÉTODOS

El estudio corresponde a una investigación documental realizada a partir de la revisión fuentes bibliográficas relacionadas con la temática referida. Se realizó un análisis cualitativo de la información obtenida con el objetivo de responder a la pregunta: ¿Qué estrategias didácticas se emplean para conducir el proceso de enseñanza-aprendizaje de la programación? No sin antes precisar por qué el interés actual por la enseñanza de la programación.

Se buscó la respuesta a partir del análisis de contenido de artículos localizados a partir de las palabras claves: proceso de enseñanza-aprendizaje de la programación de computadoras, estrategias para enseñar programación y teaching programming. Se recuperaron 45 artículos.

Como criterio de inclusión se consideró que contribuirían a dar respuesta a las interrogantes y que tuvieran fecha de elaboración después del 2014.

La irrupción de las TIC en la sociedad ha generado exigencias en la formación de los ciudadanos. Esto no es un reto actual, más bien siempre ha sido así. En las distintas épocas que ha vivido la humanidad, la incorporación de las tecnologías a los procesos educativos ha exigido nuevas destrezas, tanto en estudiantes, como en docentes, pues el proceso de enseñanza-aprendizaje tiene un condicionamiento histórico. Las necesidades sociales determinan los objetivos de dicho proceso. Se ofrecen argumentos en relación con el *interés por la introducción de la programación* en los centros escolares.

El vocablo programación en el contexto educativo, puede entenderse de diversas maneras. En este trabajo se relaciona con la palabra programar, definida por Martínez & Echeveste (2018), como *“la acción de darle instrucciones precisas a una máquina en un lenguaje que la computadora pueda entender para que ejecute y automatice alguna acción o conjunto de acciones”* (p.94). Se concibe además la Programación, como una disciplina joven.

Respecto al tema que ocupa a los autores, en no pocos países desde la década de los 70 se enseñaba

programación en los centros escolares, pues desde que surgió la computación ha existido la idea de que enseñar a programar a los niños es una experiencia provechosa para su desarrollo cognitivo y los incentiva a interesarse por profesiones que se requieren en la industria tecnológica. En los inicios de la década de los 80, Seymour Papert, creó junto a su grupo en el MIT el lenguaje LOGO, para que los niños aprendieran matemáticas en el contexto lúdico y riguroso del razonamiento lógico necesario para resolver problemas y programar una computadora (Jara & Hepp, 2016).

En Cuba también fue así (lenguaje MSX-LOGO en la primaria y MSX-BASIC en la secundaria). Varios estudios refieren que esa decisión no fue del todo satisfactoria, debido a que lejos de incentivar al alumnado y al profesorado con estas técnicas, se contribuyó a que se produjera un cierto rechazo hacia ellas. No sólo por el contenido, sino también porque el equipamiento disponible, en los cuales el acceso a la información era secuencial, ocasionaba una pérdida de tiempo muy grande para recuperar o almacenar los programas que se utilizarían en las clases.

En la década de los 90, *decayó la enseñanza de la programación*. Como señalan Jara & Hepp (2016), *“el foco de atención giró hacia el uso de computadores e Internet como apoyo a la enseñanza y el aprendizaje en forma transversal en el currículum escolar”* (p. 2)

Refieren dichos autores que ya lo importante no era *aprender sobre los computadores sino con los computadores*. Por tal razón, la mayor parte de los cursos de informática se orientaron a la utilización de las aplicaciones computacionales (procesadores de texto, planillas electrónicas, bases de datos, etc.) para resolver problemas cotidianos y asegurar las habilidades tecnológicas básicas para el uso de las TIC en el resto de las materias escolares. Situación similar ocurrió en Cuba. A partir de 1996 se rediseñó de manera integral la política sobre la Informática en las instituciones educativas, con un objetivo centrado en lograr buenos usuarios de las aplicaciones informáticas.

En la década del 2000 *vuelve a emerger con fuerza la idea de enseñar programación* pues en la sociedad actual la generación, almacenamiento y procesamiento de la información comienza a tener un auge relevante en la vida cotidiana. La economía mundial ha cambiado. La ingente abundancia de información disponible posibilita la existencia de *nuevos productos y servicios*. La transformación de la información digital en valor económico y social, ha posibilitado la creación de nuevas industrias, ha modificado otras y tiene un impacto considerable en la vida de las personas.

En 1965, según la Revista Popular Science, que circulaba en Estado Unidos, se necesitaban 500.000 personas que supieran programar. Esa situación no ha cambiado en la actualidad, todo lo contrario. En el 2020, según estimados, eran un millón las plazas laborales ofrecidas por empresas de tecnología en EE.UU. Por tal razón, ha renacido el interés por la enseñanza y aprendizaje de los lenguajes de programación en los centros educativos. Internet, las redes sociales, la telefonía móvil y la progresiva importancia de las Ciencias de la Computación, han motivado una

nueva perspectiva para entender el papel de la programación (Fábrega, et al., 2016b)

Los niños y jóvenes en su interacción con los medios se presentan en grado mayor como consumidores, no como productores de los mismos. Para cambiar esa realidad, se necesita enseñar a programar computadoras. El conocimiento de esta materia, amplía las posibilidades del uso de las tecnologías, además de que permite crear nuevos contenidos y herramientas.

En línea con esa idea Dapozo, et al. (2018), opinan que la alfabetización digital, debe integrar, entre otros elementos, nociones sobre los lenguajes de las computadoras. Tanto la *programación como el pensamiento computacional*¹ son relevantes para el aprendizaje, pues al entender su semántica y su lógica en la resolución de problemas, los estudiantes también se preparan para entender y cambiar la realidad. Lo anterior no es privativo de los que realizan una formación específica en Ciencias de la Computación, sino también, atañe a niños y jóvenes que no precisamente realizarán una formación profesional en el campo de la Informática.

Ratifica los argumentos expresados en relación con la importancia de la programación, el criterio de Fábrega, et al. (2016a), quienes opinan que en la actualidad, existe mayor conciencia de que la introducción de la enseñanza de lenguajes de programación desde la escuela, empodera a los estudiantes en el proceso de enseñanza-aprendizaje. Como señalan Vera & Argüello (2019), aludiendo a una frase de Steve Jobs quien expresó que *“todo el mundo debería aprender a programar computadoras, porque te enseña a pensar”* (p. 107) y es así, diversos estudios lo han demostrado.

En síntesis, es criterio de los autores de esta investigación que el aumento del interés por la enseñanza y el aprendizaje de la programación, tiene dos aristas fundamentales:

La necesidad social (aunque la necesidad educativa está incluida en esta, se han separado para precisar ideas): con la sociedad de la información han cambiado muchos paradigmas económicos. La información se ha convertido en un producto esencial. Existe una avalancha de datos que requiere métodos que permitan transformarlos en significado. La programación contribuye a obtener esos métodos.

La necesidad educativa: se precisa educar a los jóvenes en la comprensión de cómo funcionan las tecnologías y sus principios fundamentales, para que no solo sean consumidores sino también agentes creativos en el mundo digital. Además, el programar tiene efectos beneficiosos en el desarrollo del pensamiento.

Sobre el aumento del interés por la programación, según el razonamiento de Cathles & Navarro (2019), para hacer frente a la gran escasez de capital humano en el área digital, han surgido programas de capacitación digital privados o sin fines de lucro (bootcamps). Las empresas más importantes desean trabajar con sus egresados

¹ Es un tema que ahora mismo tiene mucha importancia en el medio educativo y se considera una habilidad a desarrollar en todo el alumnado.

(capital humano con habilidades en programación, con dominio de la inteligencia artificial y del análisis de datos). Dichas organizaciones ajustan rápidamente su estrategia de enseñanza a las exigencias y últimas tendencias empresariales. Los autores de este artículo, reconocen la valía de dichos programas, pero no deben ser considerados la solución a la necesidad de enseñar programación, pues esas instituciones son elitistas.

El desafío es cómo incorporar la enseñanza de la programación en las instituciones educativas La revisión realizada por Fábrega, et al. (2016a), revela que la tendencia mundial por integrar la enseñanza de la programación en el currículo escolar se consolida en la Eurozona, mientras que en América, Asia y Oceanía existen procesos emergentes.

Se está en plena concordancia con criterios de miembros de la Royal Society² (2017), que refieren que **diseñar un buen currículo educativo** es el primer paso para garantizar que el alumnado salga de la escuela y la universidad bien formado, para emprender carreras profesionales exitosas y convertirse en ciudadanos inteligentes y responsables. Desde la perspectiva de los autores, dentro de dicho diseño, la precisión de las competencias a lograr ocupa un lugar significativo.

Considerando tal postulado, una investigación desarrollada por San Andrés, et al. (2016), revela que un 90% de los docentes sitúa la **competencia programar software** entre aquellas de mayor importancia para los ingenieros en sistemas informáticos. Proponen la siguiente definición de la competencia profesional programación de software: **“El resultado de la integración, esencial y generalizada de un complejo conjunto de conocimientos y habilidades relacionadas con el proceso de diseño, codificación, depuración y mantenimiento del código fuente de programas computacionales aplicando técnicas y herramientas informáticas con valores profesionales, que se manifiestan a través de un desempeño profesional eficiente en la solución de los problemas propios de la Ingeniería en Sistemas Informáticos”.**

Sin dejar de considerar esa competencia, otra mirada la aportan Compañ et al. (2015), quienes expresan que una competencia fundamental es la relacionada con el trabajo en grupo. Dicha investigación da cuenta de cómo se contribuye, pues los estudiantes formados en equipo desarrollan un proyecto de programación.

Por lo anterior, los autores destacan que en la enseñanza de la programación, tan importante como el aspecto instructivo es el educativo. Se necesita que los programadores tengan valores como la adaptabilidad, la colaboración, la comunicación y la paciencia. En uno u otro caso, hay que emplear **estrategias didácticas** adecuadas para favorecer la formación de tales competencias.

Desde la perspectiva de Martínez & Echeveste (2018), **“la enseñanza de la... programación, presenta particularidades que desafían al régimen académico estándar”** (p.35). Se coincide con dicho criterio, pues programar

exige inteligencia, conocimiento, destreza, disciplina y práctica. Estos aspectos influyen en las dificultades presentadas por la mayoría de los estudiantes en estas asignaturas y por ende es un indicador negativo que afecta la calidad de su proceso de enseñanza-aprendizaje.

Los problemas que se presentan en el Proceso de Enseñanza-Aprendizaje (PEA) de la programación, han sido documentados por varios investigadores entre ellos González, et al. (2018). Por ejemplo, en Estonia el 32.2 % de los estudiantes de primer año abandonó durante 2015 carreras que se relacionaban con la informática y las tecnologías de la información. Resultados de quince universidades españolas vinculadas con la mencionada área, determinó que durante el primer año del período 2015-2016, los niveles de fracaso escolar se encontraban en 22.5 %. En Latinoamérica los resultados son similares. En un estudio en Colombia (2017), se demostró que entre 45 % y 52 % de los estudiantes que ingresaron a un programa de ingeniería en Informática lo habían abandonado. En ese mismo país, específicamente en la Universidad Simón Bolívar de Barranquilla, un estudio (2007 y 2012) que empleó la minería de datos reveló que el porcentaje de deserción se hallaba en 65 %.

El proceso de enseñanza-aprendizaje de la programación, ha sufrido algunas modificaciones a través de los años, coexistiendo varios enfoques y tendencias que están en concordancia con los distintos autores que se han dedicado a investigarlo.

En relación con otras disciplinas, no es mucho tiempo para que se haya consolidado el sistema para la enseñanza de la programación en el mundo. Así lo expresan investigadores de la Royal Society (2017), **“debido a la relativa novedad del tema, las pedagogías sobre programación están menos desarrolladas que las de otras asignaturas. Está claro, que hay docentes de dicha disciplina que están haciendo muchos esfuerzos para impartirla de la mejor manera posible, pero harían su trabajo de una manera más eficaz y los estudiantes obtendrían mejores resultados si existiera una mayor evidencia sobre los mejores métodos de enseñanza disponibles”.** (p.30)

Varios autores ofrecen alternativas para conducir el PEA de la programación. Un hallazgo de esta investigación documental es que no hay uniformidad en la forma de nombrar dichas alternativas, algunos en sus trabajos se refieren a ellas como metodología de la programación, modalidad, enfoque didáctico, pedagogía de la programación, metodología didáctica y otras denominaciones. Se decide en este estudio llamarlas **estrategias didácticas**. Se asume la definición que aparece en Díaz, et al. (2018), **“un conjunto de acciones, ordenadas y secuenciadas conscientemente por el docente, con un propósito o intencionalidad pedagógica determinado, responden a decisiones pertinentes basadas en una reflexión sobre la mejora del proceso de enseñanza y aprendizaje; y sus elementos dependen de la subjetividad, los recursos existentes y del contexto donde se desarrollan”.** (p. 78)

No es objetivo de este artículo, describirlas en toda su extensión sino señalar su esencia, orientación o la idea fundamental que subyace en ellas, a partir de los autores considerados.

2 La Royal Society es la Real Sociedad de Londres para el Avance de la Ciencia Natural.

En relación con el objetivo de la inserción curricular de la programación, se asume como señalan Compañ, et al. (2015), que enseñar programación no es en enumerar una serie de estructuras de codificación dando a conocer para que sirve cada. Como tendencia fue así.

Alea, et al. (2019), reconocen esa alternativa como *enfoque del manual o instructorista*. Fue el predominante en los inicios de la enseñanza de la programación en Cuba y en muchas partes del mundo. Se caracteriza por su énfasis en los elementos del recurso informático y no en los procesos de búsqueda en la solución de problemas. Las desventajas de su aplicación radican esencialmente en que al no centrarse en la resolución de problemas, dificulta la creación de motivaciones efectivas en el estudiante.

Años después fue otra la estrategia. Como expresan Compañ, et al. (2015), programar implica que “el estudiante aprenda a pensar, a analizar una situación y a diseñar el método de resolución más adecuado, dejando al margen el lenguaje de programación”. Sea cual sea el lenguaje utilizado, el conocimiento del lenguaje no es el objetivo en sí, sino la herramienta para expresar el algoritmo. Es fundamental que los estudiantes no se centren en las particularidades del lenguaje, tarea compleja que implica un alto nivel de pensamiento abstracto, que no es común sobre todo en los estudiantes que se inician en la programación.

Desde esa perspectiva, la experiencia de González, et al. (2015), es que los estudiantes de los programas de Ingeniería deben atender un curso básico, denominado Algoritmia y Programación cuyo aporte fundamental es el desarrollo de la capacidad de análisis y abstracción, aplicados a la consecución de solución a problemas reales. Ratifican en su estudio el alto número de abandonos y notas promedio muy bajas al final de cada curso.

Sobre la importancia de la algoritmización, Vera & Argüello (2019), consideran que *“se requiere atender el pensamiento computacional, que es lo que influye en la lógica (uso de algoritmos) que debe desarrollarse en la persona para programar y...comprender qué es la programación”*. (p. 109)

En esencia, se constata que lo importante no es la codificación en sí misma. En línea con esa idea, se ubica lo que autores cubanos llaman el enfoque algorítmico. Según Alea, et al. (2019), su aplicación transita por los siguientes pasos:

1. Se parte de un problema.
2. Se busca la solución al problema destacando los elementos como los procedimientos algorítmicos básicos ya conocidos y la aplicación de procedimientos heurísticos (principios, reglas y estrategias) y los medios auxiliares heurísticos.
3. Se modela la solución mediante una descripción algorítmica.

Astudillo & Bast (2020), refieren que entre las estrategias empleadas para que los lenguajes de programación no sean el centro de la propuesta didáctica, emplean entornos de programación por bloques, como Scratch, Alice

o Pilas Bloques, entre otros. Para ello construyen programas encastrando bloques en la secuencia correcta. Tales entornos presentan ventajas como la ausencia de errores de sintaxis propia de los lenguajes textuales y la representación visual de conceptos abstractos.

Una investigación desarrollada por Fierro (2016), da cuenta del empleo de la analogía en el proceso de enseñanza-aprendizaje de la programación en la formación inicial de profesores de Informática. Dicha autora propone un procedimiento que permite al estudiante, identificar en el problema a resolver, los rasgos que le hacen buscar relaciones de analogía con otros ya resueltos y extrapolar sus soluciones a la de la nueva situación, a través de la reformulación del problema inicial mediante preguntas de las que pueden surgir subproblemas, cuya solución constituye una parte de la solución del propuesto inicialmente.

Aunque no lo llaman así, similar es la propuesta de Amigone, et al. (2019), quienes sostienen que partir de la modificación de código existente constituye un enfoque adecuado para estudiantes sin formación previa en el área de conocimiento. Es una práctica en la cual dos programadores (driver y navigator) trabajan colaborativamente en diseño de algoritmos, codificación y prueba de programas. Pair Programming, así le llaman, demuestra ser un método efectivo para contribuir al mejoramiento de la calidad de dicho proceso.

Compañ, et al. (2015), refieren que un buen método para que reflexionen sobre cualquier estructura o tipo de datos, es colocar frente a ellos la sintaxis de dicha estructura y esperar que la interpreten, que intenten entender cómo funciona y para qué, para posteriormente explicar cómo y dónde se utiliza. Esta variante ha sido empleada con éxito por los autores de esta comunicación: en vez de proponer la solución de un problema, se propone el algoritmo para que los estudiantes expresen que se obtiene con él.

Dapozo, et al. (2018), dan cuenta de una *estrategia didáctica* implementada en la asignatura Algoritmos y Estructura de Datos I de la Licenciatura en Sistemas de Información, que consiste en el desarrollo de un conjunto de actividades, basadas en *herramientas lúdicas*, con el objetivo de estimular el pensamiento computacional. Dichos autores demostraron que incrementa la motivación de los docentes y estudiantes, dado que el aprendizaje de los conceptos básicos se da en un contexto ameno y recreativo. Estudios anteriores también confirman la eficacia de dicha alternativa entre ellos el de Compañ, et al. (2015).

En línea con la lúdica, un exhaustivo estudio desarrollado por Astudillo, et al. (2019), que se circunscribe a investigadores del proceso de enseñanza-aprendizaje de la programación en Argentina, precisa que las *estrategias didácticas* empleadas en la muestra que formó parte del estudio son: la gamificación, los juegos serios, el aprendizaje móvil y la *robótica educativa* (esta última la más usada). Dichos autores hacen una importante distinción entre gamificación y juegos serios, que puede consultarse en la fuente citada.

En relación con la robótica educativa, como estrategia didáctica, se encontraron referencias en Niño, et al. (2017),

quienes demostraron que los resultados del empleo de una Mano Robótica Didáctica, diseñada para el aprendizaje integral de programación en Arduino, fueron satisfactorios. La docente de la asignatura consideró apropiado su diseño, estructura y aplicación en el aula de clase.

La *visualización de programas* es otra estrategia didáctica que se ha empleado. Consiste en el uso de técnicas gráficas para ayudar a entender el funcionamiento de un programa de ordenador. Hay autores que distinguen tres tipos: visualización de programas, animación de programas y programación visual. Dan cuenta del empleo de esta alternativa Compañ, et al. (2015), quienes expresan que la utilización de algún sistema de visualización de programas puede ayudar al alumnado a mejorar su comprensión acerca de la ejecución de un programa. Consideran los autores, que parece más adecuada esta herramienta en una asignatura de introducción a la programación.

González, et al. (2015), desarrollaron un proyecto dirigido a lograr un curso más dinámico, de mayor atracción para los estudiantes, y disminuir los índices de deserción y elevar la calidad de los resultados. La propuesta emplea metodologías activas y específicamente o Team Based Learning (TBL).

Por su parte, Quezada & Suárez (2021), refieren que la Computación en la Nube (CN), *“proporciona un enfoque que se integra a otros elementos necesarios en el aprendizaje y permite explotar la colaboración en red”*. Las herramientas empleadas fueron: Git Hub, AWS Cloud9, DotProject. Ellas permiten la flexibilidad y capacidad de adaptación debido a que los estudiantes pueden seguir distintos ritmos en su aprendizaje, siempre contando con el adecuado acompañamiento y guía del docente

Aunque algunas de estas alternativas están incluidas en otras a las que se ha hecho mención, se considera oportuno referir la sistematización hecha por Vera & Argüello (2019), sobre numerosas investigaciones que sugieren el uso programas para aprender a programar, empleados en la búsqueda de mayor efectividad en el proceso de enseñanza-aprendizaje de la programación:

- Diseño y evaluación de algoritmos (PseInt, FREEDFD, ...).
- Entornos de programación (Eclipse, Netbeans, Visual Studio, Sublime Text, Atom, Brackets).
- Juegos (Swift Playgrounds, Mimo, Py – Learn to Code, Checkio, Codingame, Codecombat).
- Software educativo (Logo, KPL, Cantor, Mumuki).
- Herramientas visuales (Greenfoot, Codebug, Google Blockly, Alice, Robomind).
- Entornos web (Treehouse, Skillshare, Programmr, Codecademy, CodeSchool).

Por su parte Rojas, et al. (2018), emplean como estrategia didáctica, el aprendizaje basado en investigación. Refieren entre sus ventajas, que promueve las actitudes y competencias autodidactas de los estudiantes, elemento de gran importancia si se tiene en cuenta que las herramientas tecnológicas y los paradigmas que se utilizan son cambiantes y requieren una actualización permanente en los egresados.

Además, dicha estrategia moviliza en mayor grado recursos cognitivos y afectivos del estudiantado y el profesorado y los aprendizajes de los estudiantes son más sólidos, duraderos, transdisciplinares y con posibilidades de vinculación a otras situaciones semejantes. Su aplicación requiere de docentes bien preparados, motivados y con tiempo suficiente para preparar las actividades y los recursos didácticos específicos.

Los autores destacan que sea una u otra la estrategia, en la base de todas está el enfoque problémico. Al respecto Díaz, et al. (2018), señalan que los métodos que se empleen deben llevar al estudiante a identificar cuáles son las acciones necesarias y suficientes a realizar para solucionar el problema, elaborar algoritmos que permitan darle solución, codificarlos en correspondencia con el lenguaje de programación que estudie e implementarlo en el entorno de desarrollo seleccionado, teniendo en cuenta el paradigma de la programación que se emplee.

Dichas autoras emplean para la adquisición de habilidades de programación en la formación de profesores de Informática *estrategias didácticas* para:

- Generar o activar conocimientos previos.
- Facilitar la adquisición del conocimiento.
- El desarrollo de contenidos procedimentales y habilidades cognitivas.
- La adquisición de actitudes, valores y normas.

En relación con el último tipo de estrategias, se considera que hay que prestarles gran atención. Los autores han constatado en su práctica diaria que en no pocas ocasiones se descuidan. Desde el enfoque Ciencia, Tecnología, Sociedad se puede argumentar mejor el empleo de tales estrategias. Como decía el Héroe Nacional de Cuba José Martí, los adelantos de la ciencia han de ser para lograr la paz entre los hombres. Contextualizando dicha idea, se han de acometer mediante la programación, aquellos problemas que contribuyan a cumplir con tal acierto martiano. Se trata además de no concretar su cumplimiento solamente al contenido del proyecto sino también a la forma en que ha de acometerse la elaboración del programa que da solución al problema, donde se estimule la colaboración, el crecimiento humano, la solidaridad y la ética, entre otras cualidades positivas de la personalidad.

Otras estrategias didácticas empleadas tienen como centro la observación del trabajo de programadores exitosos. Al respecto Dapozo, et al. (2018), revelan que en el contexto de dificultades reconocidas sobre la enseñanza de la programación, se realizó un estudio acerca de

características comunes que tienen los estudiantes que logran sortear las dificultades y aprenden a programar. Conocer cómo actúan los estudiantes que se han destacado por su buen desempeño, aporta información para mejorar las estrategias de enseñanza.

En *resumen*, el análisis de las distintas fuentes sobre el proceso de enseñanza-aprendizaje de la programación revela que en general predomina el enfoque problémico. Las estrategias didácticas empleadas para conducir tal proceso, son más innovadoras en relación con los estudiantes (buscar su protagonismo), en el método y en los medios. Tienen su esencia, orientación o idea fundamental en:

- Que lo más importante no es la codificación en un lenguaje en particular, sino atender prioritariamente la algoritmización y el pensamiento computacional.
- La utilización de procedimientos centrados en la analogía, con lo cual se extrapolan soluciones ya dadas, a los problemas nuevos a resolver.
- Que no siempre se deben plantear problemas para que sean solucionados, sino en ocasiones es muy pertinente la realización de actividades en las cuales se plantea a los estudiantes un algoritmo y que ellos enuncien el problema al que da respuesta (formulación del problema a partir del algoritmo).
- El empleo de la gamificación, los juegos serios, el aprendizaje móvil y la robótica educativa.
- La visualización de programas, dado que el uso de técnicas gráficas para ayudar a entender su funcionamiento.
- La computación en la nube, la cual integra otros elementos necesarios en el aprendizaje y permite explotar la colaboración en red.
- El uso de metodologías activas, específicamente el aprendizaje basado en equipos o Team Based Learning (TBL).
- El empleo de programas para aprender a programar.
- El aprendizaje basado en investigación.

Vale la pena insistir en que estrategias didácticas referidas no son excluyentes, además de que su uso depende de un certero diagnóstico de los estudiantes, los docentes, los recursos y las exigencias que dimanan de los documentos legales correspondientes a cada país o institución en relación con las aspiraciones del proceso educativo. Dichas estrategias deben ser evaluadas y documentadas, pues todavía no hay plenas evidencias de cuál es la más eficaz. Como señala Compañ, et al. (2015), no hay que pensar que su utilización va a ser la solución a todos los problemas, aunque sí hay más conciencia

de que se quiere adoptar una alternativa donde el objetivo de los procesos educativos esté basado más en los procesos de aprendizaje, que en los procesos de enseñanza, pues al decir de González, et al. (2015), el autodidactismo es una respuesta a las necesidades de una formación pertinente y de calidad en las instituciones educativas contemporáneas.

CONCLUSIONES

El interés por la enseñanza de la programación en las instituciones educativas ha transitado por tres momentos fundamentales: una etapa inicial en que tuvo mucha fuerza (inició en la década de los 70 del pasado siglo), una etapa de decadencia (punto de partida en la década de los 90) y una etapa donde resurge la necesidad de su inserción curricular (desde el 2000). Esta última muy condicionada por las exigencias sociales y las necesidades educativas al respecto.

Varios estudios refieren la complejidad del proceso de enseñanza-aprendizaje de la programación y los bajos resultados académicos de los estudiantes en las asignaturas relacionadas con dicho contenido, lo cual ha motivado la obtención de variadas alternativas para lograr mayor efectividad en el mismo.

Un hallazgo de esta investigación documental es que tales alternativas se asumen con diferentes denominaciones: metodología de la programación, modalidad para la enseñanza de la programación, enfoque didáctico para la enseñanza de la programación, pedagogía de la programación o metodología didáctica para enseñar programación. Se ha decidido nombrarlas como estrategias didácticas.

A partir de la sistematización efectuada se han obtenido varias estrategias didácticas para conducir el proceso de enseñanza-aprendizaje de la programación. Están centradas en innovaciones en los métodos y los medios fundamentalmente y en ellas se estimula el protagonismo de los estudiantes. Es útil apuntar que no son excluyentes, su uso debe partir de diagnóstico profundo de los recursos humanos y materiales de la institución donde se van a aplicar.

REFERENCIAS BIBLIOGRÁFICAS

- Alea, M., Díaz, R., Santana, L. J., Díaz, G., Hurtado, F., M., B. J., & Trujillo, J. A. (2019). *Didáctica de la Informática*. Tomo I. Félix Varela.
- Amigone, F., Ramírez, R., Dolz, D. J., Parra, G., & Rodríguez, J. (2019). Enfoques metodológicos para la enseñanza de la programación en la escuela secundaria. (Ponencia). XXI Workshop de Investigadores en Ciencias de la Computación. Universidad Nacional de San Juan, Argentina.
- Astudillo, G., & Bast, S. (2020). Enseñanza y aprendizaje de programación. Hacia un estado del arte. *Virtualidad, Educación y Ciencia*, 11(20), 138-155.

- Astudillo, G., Bast, S., Segovia, D., & Castro, L. (2019). Revisión de propuestas para la enseñanza de la programación. (Ponencia) XIV Congreso Nacional de Tecnología en Educación y Educación en Tecnología (TE&ET 2019). San Luis, Argentina
- Cathles, A., & Navarro, J. C. (2019). La disrupción del talento. El advenimiento de los bootcamps de programación y el futuro de las habilidades digitales. <http://eduteka.icesi.edu.co/articulos/iadb-la-disrupcion-del-talento>
- Compañ, P., Satorre, R., Llorens, F., & Molina, R. (2015). Enseñando a programar: un camino directo para desarrollar el pensamiento computacional. *Revista de Educación a Distancia* (46), 1-15.
- Dapozo, G., Greiner, C. L., Petris, R. H., Espíndola, M. C., Company, A. M., Piragine, M. F., Stopello, M. & Lucero, I. (2018). Estrategias innovadoras de enseñanza de la programación y didácticas específicas para fomentar el pensamiento computacional. *RedUNCI -UNNE*
- Díaz, K. I., Fierro, E., & Muñoz, M. A. (2018). La enseñanza de la programación. Una experiencia en la formación de profesores de Informática. *Educación*, 27(53), 73-91.
- Fábrega, R., Fábrega, J., & Blair, A. (2016a). La enseñanza de lenguajes de programación en la escuela: ¿Por qué hay que prestarle atención? *Fundación Telefónica*. <https://www.researchgate.net/publication/312891238>
- Fábrega, R., Fábrega, J., Carreño, E., Osorio, M., & Jara, M. (2016b). Enseñanza de lenguajes de programación en la escuela, ¿Qué están haciendo en otros países? *Fundación Telefónica*. <https://www.fundaciontelefonica.cl>
- Fierro, E. (2016). La utilización de la analogía en la resolución de problemas en el proceso de enseñanza-aprendizaje de la programación. (Tesis doctoral). Universidad Central "Marta Abreu" de Las Villas.
- González, H., Peña, L., & Lemos, J. A. (2015). Propuesta para el uso de aprendizaje activo en cursos de algoritmia y programación. (Ponencia) Feria y Congreso 2015 - ÁVACO. Tolima, Colombia
- Jara, I., & Hepp, P. (2016, 08). Enseñar Ciencias de la Computación: creando oportunidades para los jóvenes de América Latina. <http://www.microsoft.com/es-es/education>
- Martínez, C. M., & Echeveste, E. M. (2018). Experiencias de programación en las escuelas. *Cuadernos de Educación*, 161(16), 92-103.
- Niño, J. A., Martínez, L. Y., Fernández, F. H., Duarte, J. E., Reyes, F., & Gutiérrez, G. J. (2017). Entorno de aprendizaje para la enseñanza de programación en Arduino mediado por una mano robótica didáctica. *Espacios*, 38(60), 23-39.
- Quezada, P. A., & Suárez, C. (2021, 02). La Computación en la Nube en el proceso formativo en Programación Web. *Revista Iberoamericana de Sistemas y Tecnologías de la Información*, (E 42), 10-19.
- Rojas, E., Naranjo, J., Campos, R., & Campos, G. E. (2018). Enseñanza de la programación: la importancia de promover actitudes autodidactas en los estudiantes. *Atenas*, 4(44), 46-53.
- Royal Society. (2017). La enseñanza de programación en los centros escolares del Reino Unido. Royal Society. <http://educalab.es/intef>
- San Andrés, E. M., Rodríguez, M. D., & Caballero, Y. (2016). La programación de software como competencia profesional del ingeniero en sistemas informáticos. *Transformación*, 12(1), 11-26.
- Vera, J. F., & Argüello, B. E. (2019). El aprendizaje de la programación de computadoras para futuros docentes informáticos. *Espirales. Revista multidisciplinaria de investigación*, 100-118.