

Enseñanza de la programación: ¿Nueva didáctica o una didáctica diferente?

Programming Teaching: New Didactics or a Different Didactics?

Ing. Erika Ascencio Jordán^{1*} (Mgs) <http://orcid.org/0000-0003-0878-6207>

Lic. Jorge E. Gil Mateos² (Ph.D.) <http://0000-0002-5384-6419>

¹Universidad ECOTEC, Guayaquil, Ecuador

²Universidad de La Habana

*Autor para la correspondencia. eascencio@ecotec.edu.ec

RESUMEN

El artículo hace un recorrido por la enseñanza de la programación de computadoras, concretamente la asignatura Fundamentos de programación, como asignatura básica en las carreras de Ciencias de Computación y otras afines, describiendo la problemática general que presenta el proceso de enseñanza-aprendizaje, reflejado en los bajos rendimientos académicos en numerosas universidades del mundo. Se relaciona el gran número de lenguajes de programación existentes en la actualidad con el creciente número de herramientas necesarias tanto para la enseñanza de la programación como para el propio hecho de programar.

Se hace énfasis en asumir transformaciones en las prácticas docentes donde el diseño curricular juegue sea fundamental, al igual que una didáctica diferente para enfrentar los problemas descritos, a partir de innovaciones en sus categorías fundamentales. Se propone el abordaje de esta cuestión con un enfoque integral y sistémico auxiliado por la inclusión de la Tecnologías de la Información y las Comunicaciones.

Palabras clave: Enseñanza-aprendizaje, programación, didáctica, herramientas, TIC

ABSTRACT

The article takes a tour of the teaching of computer programming, specifically the subject Fundamentals of programming, as a basic subject in Computer Science and other related careers, describing the general problems presented by the teaching-learning process, reflected in low academic performance in many universities around the world. The large number of programming

languages that exist today is related to the growing number of tools necessary both for teaching programming and for programming itself.

Emphasis is placed on the fact that transformations must be assumed in teaching practices where curricular design plays a fundamental role, as well as different didactics to face the described problems, based on innovations in their fundamental categories. It is proposed to address this issue with a comprehensive and systemic approach aided by the inclusion of Information and Communication Technologies.

Key words: Learning-Teaching, programming, didactics, tools, ICT

Recibido: 13/3/23

Aprobado: 15/5/23

Enseñanza de la programación

En los últimos años diferentes organizaciones gubernamentales y privadas, en muchos países desarrollados y en desarrollo, han cuestionado los contenidos que se enseñan en las materias relacionadas con la computación en la escuela obligatoria (primaria y secundaria), y las estrategias de transmisión que se seleccionan para formar a los futuros analistas, ingenieros y licenciados en computación en las universidades (Wolovick & Martínez, 2016).

Informes realizados en varios países del mundo, por comisiones especiales con referentes del área de la computación y la educación, apuntan a que lo que actualmente se enseña en las escuelas en el nombre de la computación son habilidades de usuarios de programas (Fundación Sadosky, 2016). Lejos de acercar a los estudiantes a la comprensión del mundo digital, las comisiones internacionales revelan que estas enseñanzas contribuyen a generar falsas ideas sobre la disciplina.

A pesar de los avances en las tecnologías que constituyen los mecanismos de la computación, como las supercomputadoras de bolsillo, la conectividad ubicua y de buena velocidad, los mecanismos de computación a costos muy bajos, los conceptos básicos y fundacionales de la ciencia de la computación han permanecido sin ningún tipo de modificación por más de cuatro décadas y serían esos fundamentos los que la escuela masiva debería transmitir. Lo dicho hasta aquí compete a los contenidos y materias relacionadas con la computación y especialidades

afines; pero merece un apartado específico todo lo relacionado con la enseñanza de la programación, materia fundamental en la formación de especialistas en ciencias de la computación, ingenierías en software, en sistemas, en informática, y otras especialidades cuyo núcleo es el diseño y elaboración de software, donde la programación es una materia clave. A esta área nos referiremos en lo sucesivo: la enseñanza de la programación de computadoras.

El mundo de la programación está en constante cambio, baste recordar que sus inicios están situados en la década de 1950, y hasta la actualidad existen cerca de 700 lenguajes de programación, lo que sugiere una evolución continua y muy dinámica. Los desarrolladores trabajan continuamente en nuevos proyectos, se liberan nuevas versiones de software y surgen a diario novedosas tecnologías. Todo esto, concuerda con el criterio de los autores, de que la parte principal de la formación de programadores se enmarca en el ¿qué enseñar dentro de la programación?, acompañado de una segunda pregunta tan importante como la primera: ¿cómo enseñarlo?

Una de las cosas que los estudiantes de programación deben tener clara son los beneficios de esta materia dentro del desarrollo y formación profesional, puesto que generalmente las carreras que tienen dentro de su currículo esta asignatura la justifican como que realmente es fundamental dentro del campo laboral-profesional.

Uno de los principales problemas radica en los conocimientos previos que los estudiantes tienen de la materia. Se encuentran estudiantes que han tenido poco o ningún contacto con las computadoras. Es importante mencionar que a pesar de ser jóvenes nativos informáticos dependiendo de la malla curricular de las instituciones de educación media tienen o no acceso a la computación; esta situación se repite en numerosos países; en el caso de Ecuador, con la última reforma al Bachillerato Común Unificado, no se dicta esta asignatura. Si bien es cierto que los jóvenes ahora son expertos en el uso de las Tecnologías de la Información y las Comunicaciones (TIC), entiéndase herramientas, software, etc. no necesariamente dominan la programación de computadoras. También hay que tener en cuenta el poco interés de los alumnos, en el caso de estudiantes de carreras no directamente asociadas a la computación, al no entender la utilidad de la programación de computadoras para su vida profesional.

Por otro lado en el caso específico de estudiantes universitarios de ciencias de la computación, ingeniería en sistemas, en software, en informática y otras afines la situación de la enseñanza de la programación si bien no tiene los niveles críticos antes mencionados, si se mantienen bajos

rendimientos académicos en la asignatura, donde pueden identificarse con regularidad un conjunto de causas similares a las del resto de otras carreras, así como: problemas de motivación, sobrevaloración del talento, inadecuados hábitos de estudios, estilos de aprendizaje, entendidos como un conjunto de regularidades consistentes con la forma de la actividad humana que se lleva a cabo por encima del contenido, esto es, en los dominios propios de la actividad.

La programación es una ciencia pero que mucho tiene que ver, por su creatividad, con el arte. Para Jhon Cooley, investigador del Massachusetts Institute of Technologies (MIT) programar es realmente simple, y cualquiera lo puede hacer, pero producir un buen código es un arte. La programación no es una ciencia estricta, y asevera que: sin duda hay un aspecto artístico y creativo; este autor piensa que programar es como mitad arte y mitad ciencia.

Lo visto hasta aquí nos sugiere como un eje de análisis interesante el asociado a los contenidos y materiales didácticos para la enseñanza de la programación. Uno de los textos más valiosos de las Ciencias de la Computación se titula precisamente El Arte de programar computadoras (1968), de Donald Knuth, que sentó las bases y dio nombre al análisis de algoritmos, y ha realizado numerosos aportes a varias ramas teóricas de la informática; además de esta gran obra existen otros libros de referencia para estilos de programación, tales como: The Pragmatic Programmer, de Andrew Hunt y Dave Thomas, es una colección de buenas prácticas que se deben ir adoptando con el tiempo, y debería ser consultado por cualquier persona que trabaje con código. Otros libros fundamentales son Cracking the Coding Interview de Gayle Laakman McDowell, incluye una colección de ejercicios con diferentes niveles de dificultad; Clean code: A Handbook of Agile Software Craftmanship, de Robert Martin, dedica gran espacio a explicar las técnicas y metodologías ágiles para escribir un buen código. Existen muchos y muy buenos libros para esta materia, pero tal como se puede inferir, el arte de programar requiere formación, tiempo, lecturas y mucha dedicación, y se pudiera añadir también: autocontrol para manejar las frustraciones.

Como se mencionó anteriormente el número de lenguajes de programación está alrededor de 700, lo cual indica que la variedad de estructuras, sintaxis, convenciones, etc., es enorme, súmesele a esto que existe además un variado espectro de paradigmas de la programación que pudieran influir en el aprendizaje y por consiguiente en el rendimiento académico, y que por su importancia se relaciona a continuación:

Paradigmas de la programación

Según Vaca (2000), dentro de la enseñanza de la programación, se identifican varios paradigmas que explican la composición y estructuración de la misma, a saber:

- *Paradigma Imperativo*: hace referencia a la descripción de la forma en que debe ejecutarse el cálculo, no se centra en el por qué.
- *Paradigma Declarativo*: se encarga de detallar qué es lo que se debe calcular, sin detenerse a explicar el cómo. No presenta un orden de evaluación determinado; las variables se asocian a varias definiciones y son inmutables.
- *Paradigma Funcional*: se basa en los modelos de cómputo y cálculo lambda; así como en la lógica combinatoria; presenta a las funciones como elementos de primer orden y la evaluación se realiza en base a la reducción funcional.
- *Paradigma Lógico*: se basa en la lógica de predicados de primer orden; los programas que se establecen se conforman de hechos, predicados y relaciones.
- *Paradigma reactivo*: se centra en la teoría de grafos; además de abordar la especificación de flujos de datos entre operaciones.

Estos paradigmas de programación son estilos o maneras de programación, y tienen relación cercana a las arquitecturas de software, pues esta hace referencia a la forma en que se construirá el software en cuestión. Es decir los paradigmas definen un lenguaje de programación y cómo funciona. A su vez los lenguajes de programación se clasifican en: lenguajes de máquina, lenguajes de programación de bajo nivel, y lenguajes de programación de alto nivel. Estos últimos son los que, mayormente, son enseñados en los cursos regulares de las universidades.

Herramientas en la enseñanza de la programación

Con el avance acelerado de las Ciencias de la Computación y la amplia variedad de lenguajes de programación han surgido herramientas que facilitan a los docentes la enseñanza de lenguajes o tipos de lenguajes específicos, y esas herramientas ya tienen alcance a sectores etarios diferentes; por ejemplo para los niños existen herramientas de gran efectividad en la enseñanza de la programación, a saber: ApInventor, TinkerCAD, Processing, Arduino, Code.org, Alice, Kodable, MineCraft Code Builder, CodeBug, BeetleBlocks MBlock, CodeCombat Snap!, Logo, Blockly, Scratch. Esta última de gran aceptación por la comunidad implicada, fue creada por el MIT Media Lab y es usada cada vez más en las escuelas; es probablemente la herramienta más utilizada para introducir el pensamiento computacional en los niños.

Como se puede apreciar la colección de herramientas para enseñar programación a los niños es muy amplia, variada y creciente.

Cuando se habla de enseñanza de la programación en universitarios la cuestión es más compleja, pues este grupo etario ya, muy probablemente, ha tenido experiencias anteriores en bachillerato o de forma independiente. De cualquier forma se mencionaran algunas herramientas que están consideradas como herramientas de programación muy útiles para este grupo.

Sin hacer diferencia en si son de uso gratuito o de pago, que es un tema a parte a esta investigación, son de obligada referencia las que a continuación se relacionarán; quedaría a criterio de los docentes cuál utilizar en cada caso, según las particularidades del público objetivo.

Las herramientas en cuestión son:

Para la enseñanza de la Programación web, (una variante de la programación específica en el "ecosistema" internet) se han identificado varias herramientas útiles para la transmisión de estos conocimientos, según Picie et al., (2021), estas son:

Para principiantes: GitHub, Chrome Developer Tools, Sublime Text, Marvel, Visual Studio Code, Node Package Manager, Sass, Bootstrap, Grunt, Ruby on Rails, etc.;

Y para Profesionales: Postman, Docker, Kubernetes, Sketch, NGINX, Flutter, ReactJS, Angular Vue.js, Lavarel, etc.

En la selección de estas herramientas de programación se han considerado algunos aspectos básicos: complejidad, seguridad, escalabilidad, costo, lenguajes de programación, y soporte de plataforma.

Cada una de las herramientas mencionadas responde a un propósito único: editores de código, herramientas de desarrollo de navegador, software de prototipos web, gestores de paquetes, sistemas de control de versiones (VCS), frameworks de aplicaciones web, ejecutores de tareas, herramientas de prueba de API, bibliotecas JavaScript, software de gestión de contenedores, preprocesadores CSS, y servidores web.

Como se puede apreciar el mundo del desarrollo de software es complejo y muy variado; es en ese mundo en el que se deben insertar los profesionales de las ciencias de la computación o de carreras afines, en las cuales la asignatura enseñanza de la programación es un eslabón fundamental en el propósito velado de convertir la ciencia en arte.

En la investigación denominada "La introducción de la herramienta didáctica PSeint en el proceso de enseñanza aprendizaje: una propuesta para álgebra lineal", realizada por Vargas

Ricardo & Beúnes Cañate (2019), se indica que PSeint es un intérprete de un lenguaje de programación basado en pseudocódigo; realmente es una herramienta muy completa y valiosa. El pseudocódigo es una descripción informal de alto nivel de un algoritmo, que manteniendo las convenciones estructurales de un lenguaje de programación está diseñado para la lectura humana. Está compuesto por varios módulos con diferentes funciones que se comunican entre ellos y se muestran al usuario como un todo.

La implementación de PSeint como herramienta de enseñanza de algoritmos, pseudocódigo y diagramas de flujo consta de varios pasos:

- Se lleva a cabo un análisis del enunciado del problema con el fin de obtener los datos de entrada, la forma en la que van a ser procesados para al final obtener la información deseada, siguiendo siempre la premisa: Entrada-Proceso-Salida.
- Posterior al análisis, se identifican los datos con los que se va a trabajar, se define el tipo de esos datos dependiendo de la información a manipular, ya sean numéricos o de tipo cadena de caracteres, de esta forma, esos datos serán nuestras variables.
- Se procede a llevar a cabo el algoritmo de solución del problema, consistente en una serie de pasos ordenados, finitos y congruentes que nos llevarán a la solución del problema, dicho algoritmo se escribe en forma secuencial, paso a paso, teniendo un punto de partida y un punto final.

Una vez que se tiene el algoritmo, se procede a transformarlo en pseudocódigo, el cual representará la solución del algoritmo de una forma más detallada, utilizando instrucciones en español, parecidas a un lenguaje de programación, para colocar dichas instrucciones. De esta forma los estudiantes verifican que los algoritmos que han diseñado logran el propósito deseado; genera también el diagrama de flujo. Esta herramienta es utilizada en la Universidad ECOTEC para la enseñanza de la asignatura Fundamentos de Programación, y los autores de este trabajo tienen experiencia práctica en su uso.

Como se expresó anteriormente, esta herramienta es poderosa, abarcadora y muy útil en el contexto de la enseñanza de la programación.

A pesar de la amplia variedad de lenguajes de programación es útil precisar que la amplia variedad de lenguajes de programación mencionada anteriormente, no es lo que hace más complejo el aprendizaje de la programación, pues de hecho, cada lenguaje tiene sus particularidades, pero un programador excelentemente bien formado será capaz de programar en

cualquier lenguaje, pues domina los fundamentos de la programación sea cual sea el lenguaje, y con ello será capaz de aprender a aprender. De ahí la importancia de que el proceso de enseñanza-aprendizaje de la asignatura Fundamentos de la programación sea desarrollado con un alto rigor, para ello debe considerarse muy seriamente incidir en todos sus componentes.

Acercamiento a la didáctica en la educación superior

Desde la década de los noventa organismos internacionales como la Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO) y el Centro Regional para la Educación Superior en América Latina y el Caribe (CRESALC), sugirieron la adopción de un nuevo rol del docente de educación superior, acorde con las demandas de una nueva sociedad globalizada y capaz de apoyar el desarrollo integral del estudiante; esto implica el compromiso de las Instituciones de Educación Superior (IES) de asumir una nueva visión y un nuevo paradigma para la formación de los estudiantes, basados en el aprendizaje a lo largo de toda la vida, la orientación prioritaria hacia el aprendizaje autodirigido (aprender a aprender, aprender a emprender y aprender a ser) y el diseño de nuevas modalidades educativas en las cuales el alumno sea el actor central en el proceso formativo.

Sin embargo, en la actualidad es necesario incluir a las TIC en el proceso de enseñanza-aprendizaje pues se determinan como herramientas necesarias para la adecuada adhesión de conocimientos de los estudiantes.

Una idea que resulta vital para comprender, en toda su dimensión, el alcance de esta nueva cualidad, es defendida por (Castañeda & Fernández de Alaiza, 2002; Lombillo, 2011; López, 2007), al expresar que no se trata sencillamente de incorporar las TIC como medio de enseñanza en las asignaturas y convertirlas en mera instrumentalidad; sino de hacer un uso innovador, reflexivo, integrado, progresivo y pedagógico de estas.

Castañeda y Fernández de Alaiza, (2002) indican que el mayor peligro para la educación de hoy es que se pretende hacer lo mismo que se hacía ayer, con las herramientas de hoy, con esto se estaría desaprovechando el avance tecnológico como medio didáctico en la educación. Este cambio en la educación que enmarcan Castañeda y Fernández de Alaiza, es lo que se desea lograr en el aprendizaje de Fundamentos de Programación: una estrategia o una metodología apropiada para esa enseñanza.

Una didáctica diferente

Las implicaciones de Internet y la web en la sociedad moderna son incuestionables. Friesen y Hug (2008) consideran que se han abierto múltiples canales para acceso instantáneo al conocimiento especializado que con anterioridad había sido asequible solamente en formato impreso o de forma presencial. Los medios interactivos y participativos de forma creciente van penetrando cada día en la cotidianeidad. Estos autores plantean que es necesario incluir nuevas funciones que caractericen a los medios de comunicación basados en la web. Estos deben ser concebidos como interpenetrando todos los elementos del “modelo de aprendizaje”. Los medios necesitarían ser redefinidos como ubicuos, acompañando los ambientes de clase y de la institución, y teniendo un alcance contiguo con los ambientes socioculturales como un todo. Esta afirmación cada vez más se acerca a la realidad cotidiana de hoy.

El presente apartado está ubicado en el contexto teórico de la Didáctica, y concretamente dentro de sus categorías fundamentales: objetivos, contenidos, medios, métodos, formas de organización y evaluación. Los medios, como un componente del proceso de enseñanza-aprendizaje, dependen de los objetivos y del contenido de enseñanza, y se deben planificar por el docente, en función de lo que él y sus estudiantes se propongan alcanzar (Zilberstein, 2004), (Gil, 2010). Adicionalmente están implicadas, además, las categorías métodos, y formas de organización, en tanto contienen las estrategias y metodologías, asumidas o diseñadas con el fin de lograr los objetivos del proceso de enseñanza-aprendizaje, y por último, se incluye además a la categoría evaluación del aprendizaje, como contentiva de los instrumentos garantes de que los niveles adquiridos por los estudiantes sea el deseable.

Friesen y Hug (2008) consideran que la utilización de los Nuevos Medios (NM), entendidos como los medios didácticos en la era de las TIC, necesita de una nueva didáctica que aproveche todo el potencial que estos ofrecen; las viejas prácticas no resultan eficientes y requieren ser modificadas.

Por otra parte, Fariñas aborda algunos requisitos para la enseñanza en la contemporaneidad y sugiere algunas pistas para el diseño educativo, donde destaca que:

“no existe algoritmo, método o medio de educación que permita al profesor alcanzar directa y linealmente el desarrollo deseado en los estudiantes. Sin embargo, no podemos quedarnos cruzados de brazos. Debemos determinar los límites de nuestra acción, lo que implica también estudiar nuestras posibilidades reales de influir y las de los estudiantes”. (Fariñas, 2009, p s/n).

Para esta autora es preciso que los contenidos de la educación (los cursos, actividades curriculares y extracurriculares) y los métodos de trabajo en general, cumplan determinados requisitos en aras de tales derroteros. No se trata de desterrar lo acostumbrado hasta el día de hoy en la enseñanza, pero sí de hacer otros énfasis. De cierta manera Fariñas (2009) está abogando por una nueva didáctica, o al menos por una didáctica diferente, cuando se trata de la utilización de los NM (Gil, 2010), nuevas estrategias y metodologías; concepción que es compartida por los autores de este trabajo.

Por otro lado Fariñas (2006) con relación al aprendizaje, considera que el establecimiento de un buen ambiente para el desarrollo exige estrategias como las de aprender a aprender más que el aprendizaje de contenidos específicos. “El aprender a aprender es la mejor contrapartida del diseño de ambientes estimulantes del desarrollo, donde los estudiantes tengan mayor protagonismo” (Fariñas, 2006, p. s/n).

Frente a la obviedad de la mayoría de los recursos didácticos, incitar a la lectura entre líneas, a la lectura verdaderamente crítica, a buscar los puntos ciegos en el conocimiento, puede fertilizar la capacidad para la creación. Lo obvio está al alcance y se convierte en una experiencia directa poco estimuladora para el aprendizaje (Fariñas, 2006, p. s/n); es por ello que el diseño de recursos educativos cobra gran importancia y debe estar matizado por una intencionalidad pedagógica que aproveche las características de los NM.

Por lo visto hasta aquí los autores de este artículo consideran que una línea de análisis para intentar conformar una didáctica diferente la integran los contenidos y sus estructuras; por ejemplo: los objetos de aprendizaje como estructuras de contenido, mediadoras entre el docente y el estudiante, deben ser diseñados para que este último se desenvuelva de forma protagónica y adquiera independencia en su comportamiento. Parte del protagonismo del estudiante en su actividad de aprendizaje lo constituye el hecho de que sea capaz de comprobar la calidad de sus resultados y hacer las correcciones debidas para acercarse a los conocimientos correctos. Esta comprobación está estrechamente vinculada con los objetivos de aprendizaje explícitos y con las acciones de orientación concebidas por el docente. Cada uno de estos aspectos (objetivos, orientaciones, actividades y evaluación) son componentes estructurales de los objetos de aprendizaje y es el docente quien, con su creatividad, pudiera hacer un diseño eficiente, capaz de cumplir una intención pedagógica específica.

Se asocia la calidad del aprendizaje a la eficiencia con que se enseña (Zilberstein, 2004); esto pudiera ser entendido, en contextos educativos donde se utilizan recursos educativos digitales, y nuevas estrategias y metodologías de enseñanza, como la posibilidad que tienen los docentes de diseñar recursos educativos, estrategias y opciones educativas con características particulares, que le permitan, a los estudiantes que los utilicen, asumir posiciones activas, reflexivas y conscientes. Estas intervenciones innovadoras hacia una didáctica diferente serán adecuadas cuando la participación de los estudiantes haya implicado un esfuerzo intelectual, reflexionar, valorar, suponer, llegar a conclusiones, argumentar, generar, entre otras acciones. Es por ello que se considera que para crear condiciones que potencien el comportamiento intelectual de los estudiantes se debe prestar atención a la organización de las actividades de aprendizaje.

La otra línea de análisis la conforman las categorías Métodos y Formas de organización, con ellas sucede algo similar a lo que, con la categoría Medios, pero desde una perspectiva organizacional y enfatizando en el aprendizaje colaborativo y cooperativo. Por su lado el diseño de nuevas estrategias y metodologías dinamizará el proceso de aprendizaje y situará al estudiante en una posición que le permita ser partícipe activo de su aprendizaje. Un ejemplo claro de estos tipos de estrategias podría considerarse la implementación de la Clase Invertida, entre otras, donde se cambia la forma de organización del proceso de enseñanza-aprendizaje, los roles, los tiempos, los espacios, y todo redunda en mayor aprovechamiento para los estudiantes. Por su puesto que el establecimiento de estas nuevas estrategias y metodologías requiere una implicación total del profesor, tanto en el dominio de su diseño y concepción, así como en el seguimiento del proceso.

Una de las causas de la necesidad de innovar en la didáctica está inspirada en ideas tales como las descritas por Zilberstein (2004), el cual hace un análisis de algunos de los problemas de la educación y considera que entre los de primer orden están:

- la acumulación de insuficiencias en el aprendizaje, donde los estudiantes no rebasan el plano reproductivo y no pueden operar con conceptos.
- La estimulación al desarrollo intelectual y la formación de habilidades para aprender a aprender son limitadas.

Estos problemas se hacen evidentes cuando los estudiantes tienden a aprender de forma reproductiva, es difícil reconocer donde se detienen en el aprendizaje, para poder ayudarles, y existe insuficiente ejercitación y control sistemático que permita identificar fallas y que los

estudiantes puedan aprender del error. Por otro lado, existe insuficiente atención a las formas de orientación y control de la actividad de aprendizaje.

Una vez abordados algunos aspectos de las categorías de la didáctica que en este trabajo se consideran potencialmente interesantes para la Enseñanza de la programación estamos en una mejor situación para considerar que la historia de los cursos de esta asignatura se pudiera revertir haciendo énfasis en ellas, y diseñando propuestas de carácter sistémico que estructuren contenidos, medios, formas, métodos, y evaluación, fundamentalmente en ambientes donde las TIC cobren una mayor relevancia.

Una mirada a la enseñanza de la programación desde el diseño curricular y algunas de las categorías fundamentales de la didáctica

El diseño curricular

Los autores de este trabajo consideran que el diseño curricular, tanto de toda la malla curricular de la carrera, así como de la disciplina, o a lo interno de la asignatura es el eslabón fundamental, a la hora de concebir una carrera, disciplina o asignatura.

Si nos centramos en el diseño de una asignatura se puede afirmar que un buen diseño curricular es la primera garantía necesaria, si la hubiera, en la búsqueda de los buenos rendimientos académicos. Mucho dependerá del lugar y momento dentro de la malla curricular donde se sitúe la asignatura; para ello es vital que se respeten niveles de precedencia, solapamientos y la impartición de asignaturas que coexistan en el tiempo.

Visto así, los autores de este trabajo interpretamos que, en primer lugar, el diseño curricular es primordial para el éxito del proceso de enseñanza aprendizaje. En segundo lugar se considera de gran importancia implementar de forma óptima (entendiendo óptima como extremadamente creativa, novedosa, participativa y eficaz) las categorías fundamentales de la didáctica; este análisis se abordará en el siguiente apartado.

Visión a partir de las categorías fundamentales de la didáctica

Objetivos: Es una categoría fundamental donde se establecen las prioridades del proceso, su alcance y su límite. En la medida en que su diseño sea preciso fluirán el resto de las categorías para su cumplimiento.

Contenidos: Se refiere al ¿qué enseñar? Esta categoría generalmente es la que menos problemas genera, los diseñadores tienen usualmente muy claro el conjunto de contenidos necesarios que

necesitan recorrer e interiorizar por los estudiantes. La disponibilidad de colecciones de recursos educativos (ejercicios, textos, simulaciones, etc.) alojados en repositorios institucionales es una estrategia adecuada para cualquier institución de educación superior.

Medios: Se refiere al ¿con qué enseñar? Desplegar el proceso de aprendizaje, estos han sido variados en la evolución de la enseñanza, hoy en día, sin dejar de utilizarse muchos medios clásicos (libros, manuales, revistas, etc.) se ha producido un cambio de paradigma con el desarrollo de internet, la cual es en sí misma el medio contemporáneo por excelencia. *Métodos y Formas de organización:* Se refiere al ¿cómo, de qué forma? Describiéndose la organización desde la carrera, la disciplina y las asignaturas, el colectivo de profesores, hasta el salón de clases, donde se aplicarán las estrategias metodológicas y didácticas a seguir, la implementación de ritmos, tiempos, y la definición de roles. La innovación en esta categoría le dará mucha fortaleza al proceso de enseñanza-aprendizaje.

Evaluación: Esta categoría se refiere a ¿en qué medida se cumplen los objetivos de aprendizaje? La inclusión de las TIC en la evaluación del aprendizaje no es un elemento ajeno, pues con los nuevos desarrollos en este campo se han habilitado espacios donde la evaluación de aprendizaje se puede diseñar desde la discusión entre estudiantes y docentes, lo cual le ofrece un carácter más democrático, al poder discutirse cómo evaluar de forma efectiva cada ítem de los contenidos. Existen varias herramientas para la evaluación del aprendizaje desde la perspectiva de las TIC, a saber: los talleres, tareas, foros, wikis, gamificación, y otras, pero son las rúbricas y los portafolios las que ofrecen un mayor control pormenorizado del aprendizaje.

CONCLUSIONES

La enseñanza de la programación es un problema general en las universidades del mundo, y conlleva a bajos rendimientos académicos, a segundas y terceras matrículas y hasta a deserciones. Es imprescindible diagnosticar y caracterizar a los estudiantes para actuar en consecuencia en su formación.

Existe una amplia variedad de herramientas para auxiliar a los estudiantes de programación y a los propios programadores. La variedad de lenguajes de programación no es la causa de estos resultados, sino la ausencia de didácticas efectivas.

Diseñar estrategias y metodologías específicas para esta disciplina, apoyadas en las TIC, pudieran constituir diferencias favorables en los resultados académicos de los estudiantes.

Abordar soluciones integrales sistémicas y funcionales es una vía en la ruta de resolver los actuales problemas de la enseñanza de la programación.

REFERENCIAS BIBLIOGRÁFICAS

1. Castañeda, Á., & Fernández de Alaiza, V. (2002). Aplicaciones de las Nuevas Tecnologías de la Informática y las Comunicaciones en el proceso de enseñanza-aprendizaje. *Universidad Técnica de Ambato*.
2. Fariñas, G. (2006). Psicología, educación y sociedad. Ed. Félix Varela, La Habana, Formato digital.
3. Fariñas, G. (2009). La enseñanza en la Universidad de La Habana según los requisitos de la contemporaneidad. Recuperado el 5 de enero de 2010, de:
<http://www.uh.cu/sitios/cultdoc/?q=content/La-ense%C3%B1anza-en-la-Universidad-de-La-Habana-seg%C3%BAAn-los-requisitos-de-la-contemporaneidad>.
4. Friesen, N., Hug, T. (2008). The Mediatic Turn: Exploring Concepts for Media Pedagogy. Recuperado el 01 de febrero de 2009, de:
http://learningspaces.org/n/papers/Media_Pedagogy_&_Mediatic_Turn.pdf .
5. Fundación Sadosky. (2016). Una propuesta para refundar las enseñanza de la computación de las escuelas Argentinas.
6. Lombillo, I. (2011). Estrategia metodológica para el uso integrado y progresivo de los medios de enseñanza por docentes de la Universidad Agraria de La Habana. *Tesis doctoral* . La Habana, Cuba.
7. López, A. (2007). Metodología para la asimilación de las Tecnologías de la Información y las Comunicaciones en el proceso de enseñanza aprendizaje de las Ciencias Técnicas Agropecuarias. *Revista Ciencias Técnicas Agropecuarias* , 2 (16), 63-68.
8. Picie, I., Olivares, B., López, I., Romero, C., & Reyes, L. (2021). Herramienta para la Enseñanza de la Programación usando Elementos Gráficos. *Revista Ibérica de Sistemas y Tecnologías de Información* (41), 50-62.
9. Vaca, C. (2020). *Paradigmas de Programación*. Obtenido de Universidad de Valladolid:
<https://www.infor.uva.es/~cvaca/asigs/docpar/intro.pdf>
10. Wolovick, N., & Martínez, M. (2016). Enseñar a Programar y Programar para Aprender. *Revistas* , 7 (12), 8-10.

11. Zilberstein, J. (2004). Aprendizaje desarrollador, Formato digital. Editorial Universitaria, Cuba.

Declaración de conflictos de intereses.

Los autores declaran que no tienen conflictos de intereses con este artículo.

Contribución de los autores

Los autores declaran que el artículo fue concebido, discutido, escrito, revisado y corregido con igual participación autoral.