

## **010. METODOLOGIA PARA ENSEÑAR A PROGRAMAR. UNA REVISION TEORICA**

### **Autores:**

Ing. Erika Ascencio Jordán, MSc.  
Universidad Ecotec  
Coordinador Académica,  
Facultad de Sistemas y Telecomunicaciones  
eascencio@ecotec.edu.ec

Ing. Manuel Cevallos Gamboa  
Universidad Guayaquil  
Técnico docente de Informática  
Facultad de Medicina  
manuel.cevallos@ug.edu.ec

### **RESUMEN**

Durante los últimos años se ha podido apreciar que la informática se introduce en todos los niveles y acciones de la vida diaria, y no menos en la educación.

Se enseña informática desde los niveles iniciales, ya que la nueva sociedad es nativa digital. Al inicio de la inclusión del computador en las aulas, se lo hacía para consolidar y reforzar los conocimientos de las asignaturas de la educación, tal es el caso de uso de programas educativos para reforzar ciencias naturales, matemáticas, lenguaje y en fin todas las materias que se dictan a nivel primario, esto fue escalando hasta el nivel medio donde se utilizaba ya un lenguaje de programación para diseñar pequeños algoritmos ejecutables. hasta llegar a la educación superior donde pasa de ser una herramienta de ayuda para la docencia a convertirse en una carrera profesional, al aparecer las carreras informáticas. El objetivo del artículo es revisar las metodologías que se utilizan para enseñar a programar en diferentes instituciones. El trabajo no indica el listado de contenidos que se deben dictar, sino que se centra en aspectos metodológicos, con la inclusión de experiencias y ejemplo concreto cualquier enseñanza de programación, todo esto a través de una revisión bibliográfica que determina la importancia de una buena metodología para que los universitarios desarrollen la lógica correcta de programación.

### **Palabras claves**

Metodología, programación, educación

## INTRODUCCIÓN

En los niveles de educación básica escuchar sobre informática es un tema muy atractivo para los estudiantes porque su mentalidad los lleva a pensar en juegos para completar los conocimientos adquiridos en las aulas, esto sufre cambios cuando cambiamos de nivel, ya que en la universidad el estudiante tiene que estudiar el lenguaje en el que fueron desarrollados los diferentes juegos que se utilizan en niveles inferiores de educación y concentrarse en el desarrollo de estos programas. Lo que lleva en un momento a la frustración debido a que se complica, el uso del computador ya no es como una herramienta de refuerzo sino como un medio de solución de problemas.

La enseñanza-aprendizaje es un proceso que combina dos elementos extremos, donde en un caso se encuentra la enseñanza, cuyo protagonista principal es el profesor y en el otro el aprendizaje, cuyo protagonista es el alumno. Se habla de enseñanza+aprendizaje, incluyendo un signo +, para indicar que enseñanza y aprendizaje son dos caras de una misma moneda, indisolubles, que se complementan y suman en los resultados que se obtienen. Tradicionalmente se hablaba de enseñar. Ahora se ha inclinado la balanza hacia el lado de aprender. Hay que tener en cuenta que no existe enseñanza si no hay alguien que aprenda y que lo que cree enseñar el profesor, no siempre es igual a lo que aprende el estudiante. (Llorens Largo, 2009)

Todo profesor se plantea como objetivo que sus estudiantes aprendan, y para ello son diversas las metodologías próximas a su asignatura que año tras año pone en práctica. Todo ello con una única y complicada finalidad, conseguir que sus estudiantes adquieran los conocimientos, habilidades, destrezas que esa asignatura conlleva. (López, 2009)

El profesor universitario, es seleccionado y se caracteriza por ser un experto en su área. Pero ello no necesariamente soporta que sepa cómo enseñar (López, 2009). Si buscamos el término en un diccionario, un docente es una persona que enseña (Miró, 2008) no obstante, para nosotros un docente es aquella persona que además trata de aumentar sus conocimientos no sólo sobre la materia que imparte, en este caso programación de computadores, si no sobre la forma de impartir esa materia para que los estudiantes puedan asimilar más y mejor. Es realmente importante estar motivado por la educación, cuestionándose curso tras curso las metodologías utilizadas y reflexionando acerca de cómo mejorar la docencia, siguiendo un ciclo de mejora continua.

Se han identificado metodologías de enseñanza aprendizaje que se pueden aplicar al momento de programar. Demostrando así, que es indispensable utilizar todas las técnicas y herramientas posibles que faciliten dicha labor.

Se confirma que se prefiere el trabajo a partir de ejemplos y casos particulares sobre los temas que se van a estudiar. Al mismo tiempo, se pudo verificar que ha existido una mayor dificultad en comprender los contenidos que se trabajaron de modo

deductivo que los de tipo inductivo. Ya que para aprender a programar se debe tener una mayor práctica.

Antes de continuar es necesario aclarar ciertos conceptos.

## **Aprendizaje**

El aprendizaje se define como “el proceso por el cual las personas adquieren cambios en su comportamiento, mejoran sus actuaciones, reorganizan su pensamiento o descubren nuevas maneras de comportamiento y nuevos conceptos e información”.

Una acción y efecto de aprender algún arte, oficio u otra cosa.

Adquisición por la práctica de una conducta duradera. (Real Academia Española, 2017)

El docente orienta el aprendizaje en tres terrenos o categorías principales, las cuales se sintetizan en: Aprendizaje de los saberes y su aplicación, aprendizaje de habilidades y destrezas, aprendizaje de valores y actitudes. Estas categorías coinciden con los pilares del conocimiento propuestos por (Delors, 1994) quien los percibe como aprendizajes fundamentales en el transcurso de la vida de cada persona y como las bases de las competencias del futuro. Vale la pena aclarar que Delors propuso cuatro pilares: Aprender a conocer, aprender a hacer, aprender a ser, aprender a vivir juntos. Más tarde la Oficina Regional de Educación para América Latina y el Caribe (PRELAC) propuso agregar como quinto pilar del conocimiento “Aprender a emprender”.

## **METODOLOGÍA**

Es necesario, plantear que una metodología didáctica supone una manera concreta de enseñar, método supone un camino y una herramienta concreta que utilizamos para transmitir los contenidos, procedimientos y principios al estudiantado y que se cumplan los objetivos de aprendizaje propuestos por el profesor. ¿Pero, porque escoger una metodología didáctica?

Cuando elegimos una forma de enseñar y no otra no es al azar, al contrario, esta elección depende de diversos factores. Entre ellos se menciona:

La experiencia que tiene el docente, con respecto a una asignatura con los diferentes grupos. Este proceso se denomina modelado<sup>1</sup>, porque se han tenido “modelos” de enseñanza en una disciplina.

También influyen las concepciones propias sobre lo que supone enseñar o aprender. Por ejemplo, si un docente concibe que aprender supone escuchar conceptos establecidos y que enseñar supone transmitirlos elegirá una metodología más

---

<sup>1</sup> Básicamente el modelado nos viene a decir que podemos aprender casi cualquier conducta observando como la ejecuta o lleva a cabo otra persona.

expositiva, de igual manera busca innovar, cambiar órdenes, contenidos dependiendo de la experiencia que ya tiene con grupos anteriores.

Otro aspecto a considerar guarda una relación entre la metodología que elige el profesor y sus objetivos de enseñanza. No sería la misma metodología escogida por un profesor que quiere que el alumno piense, formule preguntas, plantee problemas que aquel que pretende que el alumno conozca y reproduzca literalmente los contenidos.

Y como cuarto aspecto tiene que ver con el alumnado (su edad, intereses, nivel de conocimientos,). Por ejemplo, un docente que imparte clases en la enseñanza infantil elegirá metodologías más abiertas y participativas que aquellos que imparten clase en la universidad a gran grupo que están más centradas en la metodología expositiva.

Finalmente depende del contenido, porque si es un contenido práctico o aplicado podremos usar una metodología diferente que cuando impartimos un contenido teórico.

Cuando se empieza con un nuevo curso de clases el docente se puede enfrentar muchas veces, al desconocimiento total, por parte del estudiante, de la materia que imparte. Esto complica el proceso de aprendizaje, ya que el estudiante puede carecer de un hábito adecuado de estudio y de aprendizaje. Esta es en realidad la labor más compleja para el docente, empezar un conocimiento de cero, y a su vez, el profesor se tropieza con una serie de dificultades entre las que se puede citar en primer lugar el tiempo: el profesor no dispone del tiempo necesario para transmitir todo lo que considera que el estudiante debe saber. Otra dificultad importante a tener en cuenta es que se trata de una materia que generalmente los estudiantes no han cursado con anterioridad en las enseñanzas previas obligatorias y por tanto, desconocen la mecánica para su aprendizaje. Y añadido a todo ello, el hecho de que se trata de una asignatura que requiere que los problemas se aborden desde un pensamiento computacional, donde las soluciones deben ser representadas como secuencias de instrucciones y algoritmos. Es decir, que la esencia del pensamiento computacional es pensar como lo haría un científico informático cuando nos enfrentamos a un problema.

### **Metodología 1:**

Bajo la experiencia de (Compañ-Rosique, Satorre-Cuerda, Llorens-Largo, & Molina-Carmona, 2015), se puede acotar que una buena práctica de clase conlleva cuatro etapas:

Preparación y diseño: formulación de objetivos, organización de los contenidos, preparación de actividades para el alumnado. Esta fase sería previa a la actuación en el aula.

Introducción: ganar la atención de la audiencia, establecer relaciones con el grupo, despertar el interés, motivar hacia la tarea, presentación de objetivos, resumen general.

Cuerpo: estructuración del contenido, mantenimiento de la atención y del interés, velocidad y ritmo adecuados, expresividad, etc.

Conclusión: el objetivo es intensificar la retención, énfasis en las ideas principales, preguntas, resumen, etc.

No se puede considerar que el aprendizaje solo dependa de la motivación, porque si se decide que sólo ese es el elemento importante no lograremos el objetivo. Se consideran dos tipos de motivación interna<sup>2</sup> y externa<sup>3</sup> (Compañ-Rosique, Satorre-Cuerda, Llorens-Largo, & Molina-Carmona, 2015). Es imprescindible que haya un mínimo de motivación interna para que haya éxito en el aprendizaje. Tal y como argumenta (Jenkins, 2001), si los estudiantes no están motivados, no aprenderán.

## Metodología 2:

**Lenguaje natural:**

*Se utilizarán las variables enteras **num**, **x**, **resto***

- 1.- Inicio.
- 2.- Leer de teclado el número a investigar (**num**).
- 3.- Establecer **x** a 2.  
*(x será un número auxiliar por el que vamos a ir dividiendo **num** para ver si el resultado es o no 0).*
- 4.- Obtener el resto de la división de **num** / **x**
- 5.- Mientras (**resto** distinto de 0 y **x** menor que **num**) hacemos:
  - 5.1- Sumar 1 a **x** ( $x \leftarrow x + 1$ )
  - 5.2- Obtener el resto de la división de **num** / **x**
- 6.- Si **x** es igual a **num** entonces determinamos que **num** es un número primo, si no, entonces determinamos que **num** NO es un número primo.
- 7.- Final.

Figura 1

Una buena metodología es usar un algoritmo en lenguaje natural como el que se muestra en la figura 1, se pregunta a los estudiantes ¿qué hace?, los estudiantes responderán utilizando otras palabras lo que ya se visualiza en el algoritmo, en este caso leerán textualmente, cada línea y a final dirán que muestra un mensaje si es número primo o no, cuando nos gustaría escuchar que a la primera respuesta nos indiquen “a través de la entrada de un dato se evalúa si este es primo o no”.

---

<sup>2</sup> Desea aprender porque le gusta

<sup>3</sup> La nota y el título

Otro método puede ser el de plantear un ejercicio en el que se necesite alguna instrucción que aún no se ha estudiado, esto accionará su mente para pensar como cubrir esa carencia. Por ejemplo, tras ver solo el ingreso de datos si agrega “los datos ingresados deben estar entre un rango de 0 a 20”, esto los hará pensar en cómo lograr validar esos números de entrada, y así se introduce al nuevo tema que sería el uso de la condición (SI). Cuando se los obliga a pensar, a resolver un problema que les lleve a un nuevo concepto les ayuda a entender el porqué es así y cuáles son sus ventajas y aplicaciones. Se trata de emplear la relevancia o aplicabilidad de algo como técnica pedagógica.

Esto implica empezar a desarrollar la habilidad de resolver problemas. El pensamiento computacional se basa en resolver problemas haciendo uso de conceptos fundamentales de la informática. Aunque se van desarrollando lenguajes cada vez más cercanos al lenguaje humano, la programación de computadores no es cosa de fácil comprensión por parte de los estudiantes. A la persona que ya sabe programar le parece una tarea sencilla, pero al neófito se le complica. Es más, dominar el arte de la programación es complejo.

Con la experiencia de los docentes lo que se puede ver es que no es importante el lenguaje en que se programa sino la herramienta que se utiliza para enunciar del algoritmo.

Es obvio que el profesor está obligado a emplear un lenguaje de programación, pero lo que es fundamental es impedir que los estudiantes se centren en las particularidades del lenguaje, en sus defectos, en sus carencias, en sus bugs, etc., y desvíen la atención sobre lo importante. Esto es muy complicado, es una tarea muy difícil y que implica un determinado nivel de abstracción que los estudiantes no suelen tener. Normalmente ese nivel lo alcanza un programador cuando ya tiene cierta experiencia (Jenkins, 2002)

Se busca a través de ejercicios que los estudiantes aprendan a razonar, a pensar automatizando soluciones mediante el pensamiento algorítmico, esto es, estableciendo una serie de pasos ordenados para llegar a la solución.

A partir de esto los estudiantes se esmeran en que las cosas funcionen, a cualquier precio, con estructuras incorrectas y confusas, con el número de parches que haga falta, pero que funcione, aunque falle en ocasiones, pues si esto ocurre incorporan un nuevo parche y a funcionar. Olvidan que lo que se pretende es que aprendan a generalizar este proceso de resolución de problemas para poder ser capaces de resolver una gran variedad de familias de problemas.

### **Metodología 3:**

Para que las prácticas sean más interesantes, se empieza planteando ejercicios resueltos, otros ejercicios cortos, y otros que se puedan combinar con el aprendizaje como el cálculo de una factorial, movimiento de una ficha (a modo de juegos). Enfocados en la enseñanza, se debe recordar que el propósito es que lo que se

enseña quede en la mente y por siempre en el estudiante, para que éste en todo momento sepa cómo hacer uso de lo que se le enseñó.

Platón decía: “Ninguna enseñanza obligatoria puede permanecer en el alma ... Al educar a los niños, enséñales como si fuera una especie de juego y podrás ver con más claridad las inclinaciones naturales de cada uno”.

Siempre se ha manejado como estrategia lúdica el aprendizaje mediante juegos, no es una técnica nueva y ha tenido buenos resultados para la adquisición de aprendizajes.

#### **Metodología 4:**

Otra forma de lograr el aprendizaje es empleando el trabajo en grupo, para crear un tipo de competencia entre equipos por concluir un proyecto. Es importante recordar que en la universidad ya no hay el control que en las unidades de educación media donde controlan si va a clases, ya aquí es más autónomo todo depende del estudiante, entonces es mejor tener un tipo de evaluación continua para revisar si están aprendiendo lo que se está enseñando.

#### **Metodología 5:**

Con ésta metodología se puede ver que se comete un error, no se enseña a “pensar”; es decir, no están desarrollando la lógica de la programación. La idea es que lo fundamental al aprender a programar computadoras es desarrollar la lógica necesaria para solucionar problemas en forma algorítmica, independientemente de algún lenguaje de programación; esto es, aprender a diseñar programas (algoritmos) usando un pseudolenguaje, y no hacerlo directamente con un lenguaje. Esto es un desarrollo que viene a cooperar en la solución de una necesidad largamente experimentada por la comunidad académica de la programación de computadoras; contar con un método que permita conducir la enseñanza+aprendizaje de la programación, mediante el uso de un pseudolenguaje de diseño de programas (algoritmos). Este proceso de pensar como lo haría la computadora, se combina con todos los puntos anteriores y dan como objetivo la solución del problema planteado. Esta metodología es el resultado de la integración y adaptación de varias técnicas, como son; los conceptos y estructuras de la programación.

Dicha metodología permite diseñar programas (algoritmos) orientados a objetos, bien estructurados, bien documentados, eficaces, eficientes y fáciles de darles mantenimiento.

Adicionalmente es importante mencionar que se deben agregar otras estrategias como: una distribución correcta de tiempo, una guía práctica y la evaluación correcta en los diferentes momentos en el aula.

## CONCLUSIONES

La enseñanza a través de la resolución de problemas es actualmente el método más invocado para poner en práctica el principio general de aprendizaje activo.

Una metodología para la enseñanza basada en problemas durante el proceso docente educativo será funcional en la medida que se tenga en cuenta los otros elementos del proceso: objetivos, contenidos, métodos, medios y formas organizativas.

Aprender a programar es una tarea compleja pero esencial tanto para la adquisición de conocimiento como el desarrollo de las capacidades de pensamiento.

Se han enumerado un conjunto de metodologías que están dando buen resultado y se espera puedan mejorar la calidad del aprendizaje de programación evitando que los estudiantes desde los primeros niveles deserten de las carreras informáticas por considerar que programar es una tarea imposible.

Se debe aprovechar la primera asignatura de programación generalmente es introducción a la programación o fundamentos de programación para desarrollar habilidades de pensamiento computacional ya que implica la resolución de problemas haciendo uso de conceptos informáticos.

Enseñar programación no consiste en enumerar una serie de estructuras de programación indicando para que sirve cada una de ellas. Es más que eso, se trata de 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. Se trata de un objetivo muy complejo. Para cualquier persona diseñar la solución a un problema requiere de un esfuerzo importante de abstracción, aún más si tiene que expresarla en forma de un algoritmo.

Es fundamental que los estudiantes comprendan que la habilidad de escribir programas correctos, eficientes, bien organizados y adecuadamente documentados es un requisito esencial para cualquier titulado en informática, porque programar es mucho más que conocer un determinado lenguaje.



## **BIBLIOGRAFÍA**

Amigone, F., Kogan, P., Michelan, G., & Rodriguez, J. (2018). Edimbrujo: Definiendo de un modelo didáctico para la enseñanza de la Inteligencia Artificial en Juegos. XIII Congreso Nacional Tecnología en Educación y Educación en Tecnología.

Borfil, P., & Miro, J. (2007). Las fases del aprendizaje: Un esquema para el análisis y diseño de actividades de enseñanza/aprendizaje. XIII Jornadas de Enseñanza Universitaria de la Informática (JENUI).

Colomina, O., Compañ, P., Satorre, R., Aznar, F., Suau, P., & Rizo, R. (2004). Aprendiendo mediante juegos: experiencia de una competición de juegos. X Jornadas de Enseñanza Universitaria de la Informática (JENUI).

Compañ-Rosique, P., Satorre-Cuerda, R., Llorens-Largo, F., & Molina-Carmona, R. (2015). Enseñando a programar: un camino directo para desarrollar el pensamiento computacional. *Revista de Educación a Distancia*, 46(11).

Delors, J. (1994). Los cuatro pilares de la educación. En *La educación encierra un tesoro* (págs. 91-103). México: UNESCO.

García Molina, J. (2001). ¿Es conveniente la orientación a objetos en un primer curso de programación? VII Jornadas de Enseñanza Universitaria de la Informática (JENUI).

Hernández, C. (1997). Metodologías de enseñanza y aprendizaje en altas capacidades. Obtenido de <https://gtisd.webs.ull.es/metodologias.pdf>

Jenkins, T. (2001). The motivation of students of programming. *Proceedings of the 6th Annual Conferencia on innovation and Technology in Computer Science Education* }.

Jenkins, T. (2002). On the difficulty of learning to program. . *Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Science*, 53-58.

Llorens Largo, F. (2009). La tecnología como motor de la innovación educativa. Estrategia y política institucional de la Universidad de Alicante. *ARBOR Ciencia, Pensamiento y Cultura*.

Llorens, F. (2015). Dicen por ahí...que la nueva alfabetización pasa por la programación. *Revista de investigación en Docencia Universitaria de la Informática.*, 8(2).

López, D. (2009). Investigar en educación: guía práctica. XV Jornadas de Enseñanza Universitaria de la Informática.

Míguez, M. (2007). ¿Cómo motivar para aprender? Didáctica práctica para enseñanza media y superior, 309-340.

Miró, J. (2008). De la alquimia a la química. Revista de investigación en Docencia Universitaria de la Informática.

Quiterio Figueiredo, J. (2017). Cómo mejorar el pensamiento computacional: un estudio de caso. Education in The knowledge Society (EKS), 35-51.

Real Academia Española. (2017). Obtenido de RAE.es: <http://dle.rae.es/?id=3lacRHm>

Sheard, J., & Hagan, D. (1997). Experiences with teaching object-oriented concepts to introductory programming students using C++. Technology of Object-Oriented Languages and Systems, IEEE, 310-319.