

# Estilos y Estrategias de Aprendizaje para aprender a programar

L. G. Pérez González  
*Universidad Cooperativa de Colombia*  
*luis.perezg@ucc.edu.co*

## RESUMEN

Se describen varios estilos y estrategias de aprendizaje utilizadas en el curso de Algoritmia (Lógica de Programación) que se imparte en el primer semestre de la carrera de Ingeniería de Sistemas en la Universidad Cooperativa de Colombia.

Además de las explicaciones y fundamentos teóricos, se ofrece una descripción de los resultados observados en recientes experiencias y otras que están en desarrollo.

Se discuten y analizan las implicaciones e impacto que tienen los estilos y estrategias de aprendizaje en los estudiantes que inician programación con computadores para resolver problemas, y se suceden en tres etapas o momentos: la primera etapa denominada de diseño y alcance, fue la encargada de originar el diseño curricular y la planeación estratégica, una segunda etapa denominada de aplicación, fue la encargada de incorporar todo el proceso de análisis y diseño de las herramientas y técnicas a utilizar en la plataforma virtual para apoyar el aprendizaje de los estudiantes, y la tercera etapa denominada de evaluación, fue la encargada de mostrar los resultados de las pruebas realizadas con los grupos objeto de estudio, así como de estimar y valorar el impacto que tuvo el proyecto en los estudiantes de ingeniería de sistemas de la UCC sede Montería y se ofrecen las conclusiones y recomendaciones para futuro trabajo de continuidad o reingeniería del problema tratado.

## PALABRAS CLAVES

Aprendizaje, Estilos, Estrategias, Competencias, Algoritmos, Técnicas, Problemas, Plataforma virtual.

## ABSTRACT

Various learning styles and strategies used at the Algorithmic course (Programming Logics) currently taught in the first semester of the Computer Science Engineering degree at Universidad Cooperativa de Colombia are described.

In addition to the explanations and theoretical foundations, a description of the results observed in recent experiments and others that are in development is offered.

We discuss and analyze the implications and impact of styles and learning strategies in students who start programming computers to solve problems, and occur in three stages or phases: the first stage called design and scope, was responsible for result in curriculum design and strategic planning, a second stage called application, was commissioned to incorporate the entire process of design and analysis tools and techniques to use in the virtual platform to support student learning, and the third stage called evaluation was commissioned to show the results of tests performed with the groups under study, as well as to estimate and evaluate the impact of the project engineering students UCC systems and are offered based Monteria conclusions and recommendations for future work continuity of the treaty or reengineering problem.

## KEYWORDS

Learning Styles, Strategies, Skills, Algorithms, Techniques, Problems, virtual platform.

## INTRODUCCIÓN

*“El ingeniero debe ser un hombre de letras para mantener registro de precedentes útiles (...), un dibujante habilidoso quien puede representar los efectos deseados con dibujos a color (...), un*

*matemático que pueda utilizar la regla y el compás, puede distribuir trabajos, utilizar la óptica para considerar el efecto de la luz, y utiliza la aritmética para adicionar costos (...), un historiador para narrar eventos pasados; un diligente estudiante de la filosofía para que sea de mente abierta, sin arrogancia, imparcial, sin avaricia (...) Además debe entender los principios de la naturaleza y del flujo del agua, sin ignorar la medicina para trabajos de suministro de agua y drenajes (...) Aprender de leyes para contratos, especificaciones y pleitos (...) y finalmente estar familiarizado con cálculos de astronomía” [1]*

El Consejo Nacional de Acreditación en 1998, expresó que el concepto de calidad educativa, debe ser asociado a la realización óptima de las características específicas que le sean propias a una institución o programa académico según los campos de acción en que opere y según su propio proyecto institucional (su misión, su proyecto educativo y los propósitos y objetivos que la animan).

La búsqueda de la calidad implica no sólo atender aspectos de infraestructura o de recursos físicos, sino también de calidad académica, que permitan responder con responsabilidad al reto de la autonomía otorgada a las instituciones de educación superior. De esta manera, se precisa reflexionar sobre el cómo, para qué y a quienes se forma, ya que un modelo homogéneo de formación no sería eficiente ni pertinente para atender los requerimientos de la comunidad educativa, favoreciendo sólo aquellos que se adapten al mismo.

Desde diferentes puntos de vista y experiencias, se deduce que para el logro de los objetivos y propósitos de formación de profesionales en el área de programación utilizando SL, ésta ha de orientarse hacia la implementación de estrategias pedagógicas consecuente con las teorías del aprendizaje, las cuales señalan que el aprendizaje humano es siempre una construcción que se realiza basado en el procesamiento de la información, por lo que, la enseñanza debe “facilitar y potenciar al máximo ese procesamiento interior del alumno con miras a su desarrollo” (Flórez, R., 1994, p. 35).

En el contexto sobre los estilos y estrategias de aprendizaje se reconoce la importancia de las variables individuales ligadas al aprendizaje. En este sentido, los estilos de aprendizaje son definidos por Felder y Henríquez, (Citado por Martín, Gavilanes, A., 2004, p.108) como “los modos en que el individuo característicamente adquiere, retiene y recupera información”. También, las estrategias de aprendizaje se refieren según de la Fuente, J., y Justicia, F., (2003), al conocimiento que la persona tiene sobre: sus posibilidades y limitaciones para realizar una tarea; los requerimientos de esta última en cuanto a los pasos que se precisan para su desarrollo y los repertorios que lleva consigo; así como también la evaluación que el aprendiz realiza de su propio aprendizaje.

El actual interés por el tema de las Estrategias de Aprendizaje, es en parte promovido por las nuevas orientaciones psicopedagógicas, en investigaciones realizadas sobre el tema se ha comprobado que los estudiantes con éxito difieren de los estudiantes con menos éxito en que conocen y usan estrategias de aprendizaje más sofisticadas que la pura repetición mecánica. Es opinión común que la inversión en la mejora de las estrategias de los estudiantes es más rentable académicamente, que la mejora de las técnicas instruccionales o los materiales de enseñanza.

La principal razón para que las personas aprendan lenguajes de programación es utilizar un computador como una herramienta para la resolución de problemas, siendo tres las fases que pueden ser identificadas en el proceso de resolución:

- Fase de Identificación (qué nos plantean)
- Fase de resolución del problema
- Fase de implementación (realización) en un lenguaje de programación

Autores como Guibert, Guittet y Girard (2005), plantean que los estudiantes que se enfrentan por primera vez a la programación en su proceso de formación, presentan problemas tales como: no logran desarrollar un modelo viable o estructura que permita resolver el problema, ni describir una estrategia comprensible para la computadora o abstraer los diferentes comportamientos de una tarea en una estrategia que los integre a todos.

También, Oviedo y Ortiz (2002) en un estudio hecho en la Academia de Computación de la PIICSA, México D.F, concluyeron que existe carencia de habilidades para programar y bajo aprovechamiento docente en esta asignatura. En otros escritos, los docentes Ferreira y Rojo (2005), de la Universidad Nacional de Río Cuarto, en Argentina, confirman que los estudiantes de primer año que cursan Lógica de Programación (Algoritmia), presentan insuficiencias, dentro de las que citan: escasa destreza para desarrollar algoritmos de mediana o alta complejidad, falta de apropiación de una metodología de resolución de problemas e insuficiente experiencia en el manejo del lenguaje de programación utilizado en la asignatura.

El conocimiento de las estrategias de aprendizaje empleadas por los estudiantes y la medida en que favorecen el rendimiento en las diferentes disciplinas permitirá también el entrenamiento en las estrategias a aquellos sujetos que no las desarrollan o que no las aplican de forma efectiva, mejorando así sus posibilidades de trabajo y estudio.

Para obtener mejores beneficios de estos sistemas, en este proyecto de tesis, se consideró integrar en un sistema el concepto de estilos de aprendizaje y la educación basada en competencias. También se abordará la problemática de integrar un proceso que diagnostique si un alumno posee las competencias específicas de una asignatura, para determinar si es necesario cambiar la estrategia de enseñanza, con el fin de apoyar el aprendizaje del alumno. Este proceso será implementado en la plataforma Moodle, un sistema para crear cursos en línea.

Investigar la incidencia de las estrategias pedagógicas utilizadas para la enseñanza de la programación en los estudiantes de ingeniería de la Universidad Cooperativa de Colombia, sede Montería, es de utilidad porque:

1. A los Profesores, les sirve como diagnóstico y guía de orientación sobre la importancia del uso de estrategias pedagógicas y los beneficios que se pueden obtener en el aprendizaje de los estudiantes de ingeniería, si se utiliza adecuadamente como estrategia didáctica.
2. A los Estudiantes, les permite entender que las estrategias de aprendizajes que utilizan, van de la mano con los estilos y ritmo de aprendizajes propios de cada uno de ellos; pero que utilizando la colaboración y la cooperación pueden lograr un mayor nivel en sus competencias.
3. A la Universidad Cooperativa de Colombia, sede Montería, como institución formadora de profesionales de las ingenierías, contará con un documento que fundamenta el desarrollo de estrategias de aprendizajes para permitirles brindar un servicio de referencia a estudiantes, docentes, directores y público en general, que genere expectativas y apertura a nuevas investigaciones relacionadas con el tema.

### **MARCO TEORICO - ANTECEDENTES**

Históricamente, se conoce la referencia que muchos profesores hacen cuando utilizan diferentes lenguajes de programación de alto nivel como C, C++ o JAVA para enseñar el curso de Lógica de Programación (Algoritmia) y en el cual se han presentado grandes dificultades por parte de los estudiantes que para la etapa inicial además de aprender los conceptos relativos al diseño de algoritmos tiene que lidiar con la implementación de las soluciones en un lenguaje de programación que le exige Sintaxis, Compilaciones, Mensajes de errores (en ocasiones en otro idioma), Depuración, entre otros.

En el año 1996 (Norma Moroni y Perla Señas) [1] en su investigación “UN ENTORNO PARA EL APRENDIZAJE DE LA PROGRAMACIÓN” propusieron un entorno de programación más adecuado que permitiera a sus alumnos desarrollar los algoritmos trabajando directamente sobre la computadora evitando tener que recordar expresamente detalles del diseño estructural de los mismos. El trabajo con el editor de algoritmos sería de tipo interactivo. Ofrecería al alumno las estructuras básicas de un lenguaje algorítmico que él completará según cada caso. Cada una de esas estructuras constaría de textos fijos y de textos reemplazables. Los textos fijos quedarían en el algoritmo, y los reemplazables deberían ser sustituidos de acuerdo a su semántica.

En el año 2003 el estudiante Pablo Novara [2] de la asignatura programación 1 del programa de Ingeniería en Informática de la Facultad de Ingeniería y Ciencias Hídricas en la Universidad Nacional del Litoral, comenzó a desarrollar utilizando Borland C++ Builder un software llamado PSeInt [3] como herramienta para aprender la lógica de programación, orientada a estudiantes sin experiencia en dicha área. Mediante la utilización de un simple y limitado pseudo-lenguaje intuitivo y en español, permitió comenzar a comprender conceptos básicos y fundamentales de un algoritmo computacional.

En el 2006, los ingenieros Samer Al-Imamy, Javanshir Alizadeh y Mohamed A. Nour de Universidades de Emiratos Árabes Unidos publicaron el proyecto “On the Development of a Programming Teaching Tool: The Effect of Teaching by Templates on the Learning Process”, donde sustentaron que uno de los principales problemas relacionados con la enseñanza de un curso introductorio a la programación era la excesiva cantidad de tiempo dedicado a la sintaxis del lenguaje, lo que dejaba poco tiempo para el desarrollo de habilidades en el diseño y la creatividad de la solución algorítmica.

En julio de 2009 en la ciudad de Bogotá-Colombia, Jorge Alberto Villalobos Salcedo ingeniero de sistemas y computación de la universidad de Los Andes, presenta el “CUPI2 – UNA SOLUCIÓN INTEGRAL AL PROBLEMA DE ENSEÑAR Y APRENDER A PROGRAMAR” [4], donde considera entre otros, lo siguiente:

- Un modelo pedagógico adaptado al perfil de los estudiantes actuales.
- Un modelo de evaluación orientado a la verificación de las habilidades que se han debido generar en los alumnos, que tenga en cuenta los diferentes aspectos que hacen parte de la tarea de programar.

En el año 2012 el Ingeniero Robert S. Moreno Mosquera, candidato a Magister en Software Libre realizó unas encuestas a 2 grupos de sus alumnos para verificar el estado en el cual finalizaron los cursos de lógica de programación en ingeniería Civil de la Universidad Tecnológica del Choco, utilizando para ello la herramienta SEUPRO en uno

de los grupos para determinar mediante comparación los resultados del aprendizaje.

Ante la problemática que se manifiesta en la interpretación de la programación y sus conceptos, se destaca el análisis, entre otros:

- Características de los estudiantes en cuanto a su estilo de aprendizaje y los conceptos menos comprendidos en la programación.
- Metodología empleada por los estudiantes para plantear la solución algorítmica a los problemas.
- Funcionamiento y calidad del diseño del algoritmo.
- Utilización de métricas por los profesores para la calificación de los algoritmos propuestos.

Las estrategias deben ser diseñadas de modo que estimulen a los estudiantes a observar, analizar, opinar, formular hipótesis, buscar soluciones y descubrir el conocimiento por sí mismos.

Inicialmente, en el curso de Lógica de Programación, se debe atender la claridad o entendimiento que tiene los estudiantes sobre los principales conceptos teóricos, como por ejemplo: qué es un problema, cuáles son las partes componentes de un problema, qué es un algoritmo, qué es un programa. Si se logra cumplir cabalmente esta primera etapa con los estudiantes, se puede proceder a realizar el análisis de cada una de las partes componentes de un problema, utilizando para ello los siguientes pasos en su orden:

- Definición del problema
- Análisis del problema
- Diseño de un algoritmo de solución
- Codificación
- Prueba y depuración
- Documentación y Mantenimiento.

El eminente y reconocido matemático, George Pólya<sup>1</sup>, creó un “Plan” consistente en un conjunto de cuatro pasos y preguntas que orientan la búsqueda y la exploración de las alternativas de solución que puede tener un problema. La finalidad

---

<sup>1</sup>  
[http://es.wikipedia.org/wiki/George\\_P%C3%B3lya#Rese.C3.B1a\\_biogr.C3.A1fica](http://es.wikipedia.org/wiki/George_P%C3%B3lya#Rese.C3.B1a_biogr.C3.A1fica)

del método es que la persona examine y remodele sus propios métodos de pensamiento de forma sistemática, eliminando obstáculos y llegando a establecer hábitos mentales eficaces; lo que Pólya denominó pensamiento productivo. A pesar de que su libro “How to Solve It” (Cómo plantear y resolver problemas) fue escrito en 1945, su pensamiento y su propuesta todavía siguen vigentes.

Dentro de las herramientas más utilizadas en la introducción a la programación, se destacan las siguientes:

- PSEINT: es un sencillo programa de comprensión asequible para los que no cuentan con conocimientos informáticos avanzados. Con la interpretación de los pseudocódigos se puede aprender programación, mediante un sencillo método de aprendizaje, haciendo ameno y fácil de comprender las escrituras de algoritmos.
- BASIC-256: es un programa con el que se puede introducir a los estudiantes a la programación de Basic de una manera muy simple y divertida. El programa ha sido diseñado para niños, a partir del uso de comandos muy simples, por lo que no es nada técnico que ayude a personas profesionales.
- ALICE: es una herramienta que es mucho más que un programa informático, se trata de un extenso proyecto cuyo objetivo es acercar el lenguaje de programación a los niños. Con esta herramienta la programación se alcanza a partir del descubrimiento a través de personajes animados que son los protagonistas de este software.
- MIT SCRATCH: es un editor de multimedia, desarrollado para que la programación pueda ser introducida en edades muy tempranas, sin asustar a los alumnos con complicados lenguajes, sino haciéndolo de un modo interactivo y muy visual. Deja a un lado el puro lenguaje de programación para trabajar a partir de bloques de colores, con los que se podrán crear las estructuras básicas, como bucles y secuencias, pudiendo editar posteriormente los parámetros de cada una de esas funciones.
- SLE: es un lenguaje de programación diseñado para estudiantes de informática, que también puede dar juego para usuarios más experimentados. Por lo tanto, es una manera de empezar en el mundo de la programación, comprender su funcionamiento para introducirse en lenguajes más potentes, y destaca en él, las funciones y subrutinas ya creadas, de forma que algunas tareas de programación se ven facilitadas, y la incorporación de un conjunto abundante de operadores aritméticos y relacionales lógicos.
- SimuProc: es un programa que simula un procesador con el cual podrás aprender las nociones básicas para empezar a programar en lenguaje ensamblador. Permite observar todo el proceso interno de ejecución del programa a través de cada ciclo del procesador.
- Guido van Robot: es un lenguaje de programación el cual ha sido desarrollado en Python, diseñada para introducir a los principiantes en los lenguajes de programación más básicos (secuencias, condiciones, bucles, procedimientos, etc).
- DFD (Diagrama de Flujo de Datos): es un software diseñado para construir y analizar algoritmos. Permite crear diagramas de flujo de datos para la representación de algoritmos de programación estructurada a partir de las herramientas de edición que para éste propósito suministra el programa.
- LPP (Lenguaje de Programación para Principiantes): fue diseñado con la idea de facilitar el proceso de enseñanza-aprendizaje de un lenguaje de programación, este contiene la mayoría de instrucciones que tienen los lenguajes de programación. Es una forma fácil de aprender ya que se trata de una forma de programación en nuestro propio lenguaje.

### **Estilos de Aprendizaje**

El término estilos de aprendizaje es definido por Kafee (1979) como “los rasgos cognitivos, afectivos y fisiológicos que sirven como indicadores relativamente estables, de cómo los alumnos perciben, interaccionan y responden a sus ambientes de aprendizaje” (Contijoch, M., 2006). Para Smith

(1988, p.24, Gallego, D y Alonso C., 2005, p. 8) los estilos de aprendizaje son “los modos característicos por los que un individuo procesa la información, siente y se comporta en las situaciones de aprendizaje”. Así también según Talavera (2001) los estilos de aprendizaje son “la predisposición de la persona a actuar de una determinada manera frente a los nuevos aprendizajes (p.2). Además, Gregorc afirma que el estilo de aprendizaje consiste en comportamientos distintivos que sirven como indicadores de cómo una persona aprende y se adapta a su ambiente (1979, referenciado por Gallego, D y Alonso C., 2005).

Autores como Aguirre, Cancino y Loaiza, (2005), asumen que los estilos de aprendizaje son relativamente estables, y en consecuencia pueden cambiar. “Los alumnos conforme avanzan en su proceso de aprendizaje descubren mejores formas o modos de aprender, por lo tanto, van a variar su estilo, además dependerá de las circunstancias, contextos y tiempos de aprendizaje que tengan que enfrentar. Los estilos de aprendizaje se centran en las fortalezas y no en las debilidades. En tal sentido, no existe correcto o incorrecto estilo de aprendizaje” (p.12).

### **Origen del Concepto Estilos de Aprendizaje**

A través de los textos, el término "estilo" se define como una cualidad interna en el comportamiento del individuo; "una cualidad que persiste a pesar de que el contenido de la información cambie" (Fischer & Fischer, 1979, Contijoch, M., 2006, p.2). Hace muchos años, Lewin (1935, citado por Hederich, C. 2004) utilizó la noción de estilo como una expresión de la personalidad, manifestada en una disposición al uso de ciertas habilidades cognitivas. “El estilo se entiende como un conjunto de regularidades consistente en la forma de la actividad humana que se lleva a cabo por encima del contenido, esto es, en los dominios propios de la actividad” (Hederich, C. 2004, p. 10).

Según estudios existentes hacia la década de los 50, las investigaciones para diferenciar los tipos de personalidad se orientaron sobre los estilos cognitivos, que “se refieren a las diferencias individuales en la manera de percibir, organizar, analiza o recodar información o experiencias” (Contijoch, M. 2006, p.1). Los estilos cognitivos

son constructos teóricos que explican lo que ocurre en la mente del sujeto cuando éste elabora una respuesta frente a los estímulos ambientales, los procesa y se enfrenta a la realidad.

El origen del concepto estilos de aprendizaje está vinculado al interés por establecer las causas que distinguen a una persona de otra; debido a que cada persona utiliza su propio método o estrategia para aprender y aunque las estrategias varían según lo que se quiere aprender, cada persona tiende a utilizar ciertas preferencias que definen un estilo de aprendizaje. Cada estilo de aprendizaje comporta sus propias fortalezas y debilidades, son flexibles y factibles de modificarse (Contijoch 2006).

### **Clasificación de los Estilos de Aprendizaje**

Las distintas clasificaciones existentes de estilos de aprendizaje, en su mayoría, toman como punto de partida la forma de percibir la información y/o la manera de procesarla.

La clasificación realizada por Jung (1923, citado por Silver, H., 2000) sobre las dimensiones de personalidad de acuerdo con las funciones cognitivas percepción –formas de captar la información, sentido e intuición - y juicio - procesamiento de la información captada pensamiento y sentimiento -, fue la base para derivar la tipificación orientada al proceso de los siguientes estilos de aprendizaje:

- Los aprendices Sensoriales-Racionales o aprendices maestros, se caracterizan por seguir instrucciones y realizar actividades de manera secuencial.
- Los aprendices Intuitivos-Racionales o aprendices comprensivos, tienen preferencia por la intuición y el pensamiento y en consecuencia tienden a percibir y comprender de manera generalizada un evento.
- Los aprendices Sensoriales-Emocionales o aprendices interpersonales: La inclinación hacia los sentidos y los sentimientos para captar y procesar la información, genera que los aprendices se orienten preferentemente hacia las actividades grupo.
- Los aprendices Intuitivos-Emocionales o aprendices auto expresivos: tienen preferencia por captar y usar la información

a través de la intuición y el sentimiento respectivamente.

## LA EXPERIENCIA: ANÁLISIS Y DISEÑO DEL SISTEMA DE APRENDIZAJE

Literalmente, se denomina sistema a la “suma total de partes que funcionan independientemente pero conjuntamente para lograr productos o resultados requeridos, basándose en las necesidades”. (Kaufman).

Según el diccionario de la Real Academia Española, sistema “es el conjunto de reglas o principios sobre una materia racionalmente enlazados entre sí, o el conjunto de cosas que ordenadamente relacionadas entre sí contribuyen a determinado objeto”.

En el contexto social actual, en la gran mayoría de las veces distintos autores coinciden en definir un sistema como “un todo estructurado de elementos, interrelacionados entre sí, organizados por la especie humana con el fin de lograr unos objetivos. Cualquier cambio o variación de cualquiera de los elementos puede determinar cambios en todo el sistema”.

### Diseño curricular y la planeación estratégica del curso.

En la Universidad Cooperativa de Colombia, para el Programa de Ingeniería de Sistemas se definió como una de las competencias transversales la referida a “Modelar matemáticamente procesos a partir de la representación de los fenómenos naturales para resolver problemas relacionados con materiales, estructuras, máquinas, dispositivos y sistemas que en forma segura logren el objetivo esperado”, y una de sus unidades plantea la competencia “Resolver problemas relacionados con materiales, estructuras, máquinas, dispositivos, sistemas y procesos para satisfacer necesidades sociales haciendo uso de la lógica matemática”, con la finalidad de contribuir al desarrollo de tres elementos específicos:

1. Reconocer las variables de un problema.
2. Interpretar el comportamiento de las variables que inciden en un problema.

3. Diseñar alternativas de solución a problemas.

La propuesta pedagógica está enmarcada según la teoría de que nuevas metodologías o experiencias asistidas por TIC pueden ser aplicadas para motivar e involucrar a los estudiantes a aprender de manera entusiasta temas o conceptos que pueden percibir no son el foco principal de sus estudios (Gold, 2010).

En la investigación realizada, se generalizan las experiencias desarrolladas en un curso de Algoritmia y se implementaron estrategias y escenarios alternativos para el desarrollo de las actividades. El objetivo general de estas estrategias estuvo encaminado a lograr que los estudiantes se involucraran activamente con su propio aprendizaje al mismo tiempo que fueran parte de una aprendizaje colaborativo o participativo basado en la metodología MICEA<sup>2</sup>, propia de la universidad.

A partir de lo anterior, se diseñó el Plan de Curso para “Lógica de Programación (Algoritmia), considerando como importante tratar los temas siguientes como saberes esenciales:

- a) Definición de Algoritmo.
- b) Tipos de Algoritmos.
- c) Expresiones aritméticas y lógicas.
- d) Evaluación de expresiones.
- e) Variables, constantes y operadores.
- f) Técnicas de representación de algoritmos y diagramas de flujo.
- g) Uso de variables.
- h) Prueba de Escritorio
- i) Representación de entrada y salida de datos.
- j) Metodología para la solución de problemas:
  - Definición del Problema;
  - Planeación de la Solución: a) Definición de Variables, b) Proceso Algoritmo: a) Diagrama de Flujo, b) Diagrama Estructurado (Nassi-Schneiderman), c) Pseudocódigo
- k) Sentencia de entrada y salida.
- l) Sentencias de decisión:

<sup>2</sup> Metodología Interdisciplinaria Centrada en Equipos de Aprendizaje.  
[https://books.google.com.co/books?id=jANX2nEQthsC&pg=PA54&lpg=PA54&dq=micea%2Bcrisanto&source=bl&ots=9m1eTlSJCR&sig=-JtVjyuN5HW6s7MsFgtuSq\\_FDR0&hl=es&sa=X&ei=kyO4VnysA8vgggTb54KQCw&ved=0CCQQ6AEwAg#v=onepage&q=micea%2Bcrisanto&f=false](https://books.google.com.co/books?id=jANX2nEQthsC&pg=PA54&lpg=PA54&dq=micea%2Bcrisanto&source=bl&ots=9m1eTlSJCR&sig=-JtVjyuN5HW6s7MsFgtuSq_FDR0&hl=es&sa=X&ei=kyO4VnysA8vgggTb54KQCw&ved=0CCQQ6AEwAg#v=onepage&q=micea%2Bcrisanto&f=false)

- Decisión SI a) Si Simple, b) Si Completo, c) Si Anidado;
  - Decisión Cuando (Múltiple).
- m) Sentencias Repetitivas:
- Ciclo Repetitivo Mientras Que,
  - Ciclo Repetitivo Para,
  - Ciclo Repetitivo Hacer Hasta
- n) Arreglos: Unidimensionales (Vectores), Bidimensionales (Matrices).
- o) Funciones o Subrutinas: Paso de parámetros por Valor, Paso de parámetros por Referencia.

El curso tiene carácter presencial, y se desarrolla tanto en Aula como en Laboratorio, este último para la finalidad de las actividades prácticas que deben cumplimentar los estudiantes.

Para la evaluación, como uno de los componentes del proceso de aprendizaje se utilizó como técnica la Rúbrica, la cual fue planificada para diferentes momentos y actividades.

### Diseño de una herramienta pedagógica basada en software libre para la determinación del nivel de aprendizaje en la solución de problemas.

Se diseñó e implementó un WebQuest que de manera general le permitió al estudiante conocer su estilo de aprendizaje para que pudiera adoptar nuevas estrategias conducentes a mejorar el proceso de aprendizaje. Un ejemplo de esta aplicación se muestra en la siguiente figura (ver Figura 1)



Figura 1. WebQuest – Estilos de Aprendizaje - Página principal de acceso

Además, se diseñaron diferentes aplicaciones WebQuest para el entrenamiento y refrescamiento de aspectos importante del curso de Lógica de Programación (Algoritmia), basado en los saberes esenciales planificados en el Programa de Curso. A manera de ejemplo de las aplicaciones se muestran

algunas imágenes resultantes de la webquest “Variables, Constantes y Operadores” (ver Figura 2)



Figura 2. WebQuest – Variables, Constantes y Operadores - Página Principal

### Diseño de un curso e-learning que evidencie la aplicabilidad desde el punto de vista didáctico de estrategias de aprendizaje.

Basado en el diseño de investigación cuasi experimental, se implementó la herramienta tecnológica MOODLE a un grupo de estudiantes de primer semestre en el curso de Algoritmia, mientras que el resto de estudiantes de este mismo curso paralelo no fue sometido a este recurso TIC. Junto con ello, se adaptaron estrategias y evaluaciones que buscaron establecer los niveles de las habilidades y aprendizaje, aplicando una previa a los 2 grupos, antes del uso de la herramienta TIC, y posteriormente una al final del proceso de implementación.

Mediante la plataforma, se pueden planificar y desarrollar una serie de actividades (Cosano, 2007). Por ejemplo:

- Tareas. Permite la asignación de un trabajo que se deberá preparar en algún medio digital (en cualquier formato) y remitirlo.
- Chat. Permite que los participantes mantengan una conversación en tiempo real.
- Foros. Es donde se dan la mayor parte de los debates, y pueden incluir la evaluación de cada mensaje por los compañeros.
- Glosarios. Permite a los participantes crear y mantener una lista de definiciones.
- Cuestionarios. Permite al profesor diseñar y plantear cuestionarios en diferentes formatos (opción múltiple, falso/verdadero y respuestas cortas).



- SCORM. Un bloque de material web. Lecciones. Proporciona contenidos de forma interesante y flexible.
- Encuestas. Un conjunto de instrumentos verificados que se han mostrado útiles para evaluar y estimular el aprendizaje en contextos de aprendizaje en línea.
- Wikis. Un Wiki posibilita la creación colectiva de documentos en un lenguaje simple de marcas utilizando un navegador web.
- Talleres. Es una actividad para el trabajo en grupo con un vasto número de opciones.
- Correo electrónico. Un método de comunicación simple entre usuarios, tanto tutores como alumnos, mediante el envío de mensajes de correo electrónico.

Con el uso de estos recursos se establecen cuatro tipologías diferentes de aprendizaje (Cobo y Pardo, 2007).

1. Aprender haciendo (learning-by-doing): Para este tipo de aprendizaje resultan de especial utilidad aquellas herramientas que permiten al estudiante y/o docente la lectura y la escritura en la Web, bajo el principio de “ensayo-error”.
2. Aprender interactuando (learning-by-interacting): Una de las principales cualidades de las plataformas de gestión de contenidos es que además de estar escritas con hipervínculos, ofrecen la posibilidad de intercambiar ideas con el resto de los usuarios de Internet.
3. Aprender buscando (learning-by-searching): Uno de los ejercicios previos a la escritura de un documento, trabajo, ensayo o ejercicio, es la búsqueda de fuentes que ofrezcan información sobre el tema que se abordará.
4. Aprender compartiendo (learning-by-sharing): El proceso de intercambio de conocimientos y experiencias permite a los educandos participar activamente de un aprendizaje colaborativo.

Lo primero que se logra cambiar está relacionado con la metodología del curso, y consistió en la utilización del software libre PSeInt<sup>3</sup> en las clases lo que permitió a los estudiantes aprender la lógica de

programación, aún aquellos sin experiencias en el área, mediante la utilización de un simple pseudo-lenguaje intuitivo y en español que les permite comenzar a comprender conceptos básicos y fundamentales de un algoritmo computacional. Esto contrasta de manera notable con el modo tradicional de enseñar a programar mediante lenguajes de programación ya que los estudiantes pueden elaborar su solución utilizando conceptos tales como constantes, variables, asignaciones, comparaciones lógicas, etc y ver de forma inmediata los resultados.

En esta etapa también se logra cambiar el concepto de evaluación del curso, incorporando la estrategia de la construcción de un proyecto que permita integrar desde un inicio lo que se aprende. Los estudiantes bajo la orientación y seguimiento del profesor definen su propio proyecto, que no es más que un programa algo más extenso y complejo y que resuelve algún problema relacionado con un contexto real de aplicación tecnológica, e incorporando el uso de la plataforma virtual Moodle como elemento que soporta y apoya las actividades presenciales según el programa de curso, a través de espacios de consulta (foro) y participación colaborativa y otros materiales complementarios. (ver Figura 3).



Figura 3. e-Aprende – Plataforma virtual en Moodle para el curso de Algoritmia.

Se realizó la programación del curso de Algoritmia, por semanas con diferentes actividades de apoyo y con el uso herramientas que brinda la plataforma (ver Figura 4).

<sup>3</sup> <http://pseint.sourceforge.net/>



Figura 4. e-Aprende – Plataforma virtual en Moodle para el curso de Algoritmia. Programación por semanas.

## CONCLUSIONES

A continuación se describe la experiencia a partir de los cambios observados en el curso y los resultados obtenidos con la aplicación de las diferentes estrategias de aprendizajes y las herramientas aplicadas.

Los resultados del trabajo aportaron nuevos indicadores en el quehacer de profesores y estudiantes del Programa de Ingeniería de Sistemas de la sede Montería de la Universidad Cooperativa de Colombia, por cuanto a los Profesores les sirve como diagnóstico y guía de orientación sobre la importancia del uso de estrategias pedagógicas y los beneficios que se pueden obtener en el aprendizaje de los estudiantes de ingeniería, si se utiliza adecuadamente como estrategia didáctica. A los Estudiantes les permitió entender que las estrategias de aprendizajes que utilizan, van de la mano con los estilos y ritmo de aprendizajes propios de cada uno de ellos; pero que utilizando la colaboración y la cooperación pueden lograr un mayor nivel en sus competencias; su esfuerzo se enfoca en cómo procesa la información, conoce en cada momento su actuar, lo perfecciona y lo contextualiza. Finalmente, a la Universidad le queda un documento que fundamenta el desarrollo de estrategias de aprendizajes para permitirle brindar un servicio de referencia a estudiantes, profesores, directores y público en general, que genere expectativas y apertura a nuevas investigaciones relacionadas con el tema.

### Aspectos y Logros significativos

- En cuanto a los estilos de aprendizaje, se encontró que existen diferencias en las preferencias por los estudiantes, y aunque la

mayoría tiene un estilo Moderado, otros en menor proporción prefieren trabajar con información en forma Visual o Sensitiva.

- En cuanto a la variable estrategias de aprendizaje, la mayoría de los estudiantes usan estrategias de Adquisición que les permiten el reconocimiento literal de la información (repetición y lectura), en tanto que las estrategias que les permitiría desarrollar aprendizaje con autonomía las utilizan en un nivel bajo como son las de Codificación; elaboración, conciencia de la funcionalidad de estrategias y el control de ansiedad, perteneciente a las escalas de Recuperación y Apoyo respectivamente.
- Al tener sistematizada la información resultante de la investigación, el profesor y las directivas tienen la posibilidad de rediseñar las estrategias de enseñanza adecuándolas al estilo y las estrategias de aprendizaje que se han detectado y que deben ser consideradas de necesarias para que el estudiante las ejercite y aplique.
- Pese a la existencia de una gran variedad de investigaciones e intervenciones pedagógicas referidas al proceso de resolución de problemas este continua siendo un elemento esencial en los aspectos teóricos, metodológicos y epistemológicos de la asignatura, que garantizan recursos para que los estudiantes puedan enfrentar y responder a la diversidad de situaciones del entorno sociocultural.
- Las herramientas y objetos virtuales diseñados para apoyar las asignaturas del componente de programación no deben centrarse únicamente en la sistematización de contenidos, sino que también deben contener asociaciones entre las diferentes herramientas de tecnología de información que permitan garantizar la interacción docente estudiante y apoyar las necesidades de aprendizaje de la lógica de programación y la aplicación de algoritmos de programación en contextos organizacionales reales.
- Los estudiantes adoptaron un conjunto de características como: “Interdependencia positiva” en el grupo donde participaron, “Interacción” entre los integrantes del grupo que los enriquece mediante la retroalimentación de ideas, “Aportación individual” a partir de la tarea asumida y el espacio para compartirla,

“Habilidades personales y de grupo” mediante la participación, el liderazgo y la coordinación de actividades.

- Se concretó una visión más amplia del aprendizaje significando la distribución y la colaboración que va más allá de los paradigmas tradicionales de la enseñanza.

## RECOMENDACIONES A FUTURO

Sin duda alguna esta es una nueva línea de investigación que debe continuar siendo objeto de estudio y de atención. Es clara y motivante la oportunidad que se le presenta al estudiante para comprender con fundamento y responsabilidad el rendimiento académico que logra en sus diferentes etapas a partir de las características psicopedagógicas que intervienen en el proceso de aprendizaje.

Después de finalizada la investigación nos permitimos plantear las siguientes recomendaciones:

- Ampliar y diversificar los instrumentos que ayudan a la recolección de la información, prestando especial atención a las estrategias didácticas y pedagógicas que se utilizan en la conducción del aprendizaje de la asignatura Lógica de Programación (Algoritmia).
- Los profesores tendrán que planear el sistema de evaluación acorde a los diferentes estilos de aprendizaje de sus estudiantes, es implementen una variedad de estrategias e instrumentos de evaluación que tengan en cuenta las diferencias individuales de sus estudiantes y que les garantice mayores posibilidades para alcanzar un mejor desempeño académico.
- Crear un grupo integrado por profesores de la asignatura Lógica de Programación (Algoritmia) para no solo producir nuevas herramientas y recursos, sino prepararlos para evaluar, asesorar y divulgar las experiencias de aprendizaje de los estudiantes.

## REFERENCIAS BIBLIOGRÁFICAS

ABELLA, A., SÁNCHEZ, J., y SEGOVIA, M. A. Libro Blanco del Software Libre en España. 2004

ALONSO, G. C., y GALLEGO, D., HONEY, P. (1999). Los estilos de aprendizaje: procedimientos de diagnóstico y mejora. Bilbao: Mensajero, D.L.

GALLEGO, D., y ALONSO C., (Agosto, 2005). Los estilos de aprendizaje: Enseñar en el Siglo XXI. Conferencia presentada en el Congreso por una Educación de Calidad en el Caribe Colombiano, Agosto, Barranquilla.

CAIRÓ, O. Metodología de la programación: algoritmos, diagramas de flujo y programas. 3ra edición. AlfaOmega.

CAMPBELL, D. y Stanley, J. (2005). Diseños experimentales y cuasi experimentales en la investigación social. Buenos aires: Amorrouto.

COBO, C. y PARDO, H. (2007). Planeta Web 2.0. Inteligencia colectiva o medios fast food. Grup de Recerca d'Interaccions Digitals, Universitat de Vic. Barcelona / México: Flacso México. Recuperado el 12 de agosto de 2014 desde [www.flacso.edu.mx/planeta/blog/index.php?option](http://www.flacso.edu.mx/planeta/blog/index.php?option)

CONTIJOCH, E. M. (2006). Relación entre los estilos de aprendizaje, estrategias de aprendizaje y aprendizaje auto dirigido. Revista LEA (Lengua en aprendizaje auto dirigido). Revista electrónica de la mediateca del CELE-UNAM. Recuperado el 20 de abril de 2014 desde <http://cad.cele.unam.mx:8080/RD3/0001a03-A.html>

COSANO, F (2006). La plataforma de aprendizaje Moodle como instrumento para el Trabajo Social en el contexto del Espacio Europeo de Educación Superior. Universidad de Málaga. Recuperado el 20 de agosto de 2014 desde [www.dialnet.unirioja.es/servlet/fichero\\_articulo?codigo](http://www.dialnet.unirioja.es/servlet/fichero_articulo?codigo)

FLÓREZ, R., (1994). Hacia una pedagogía del conocimiento. McGraw Hill, Santafé de Bogotá, D.C Colombia.

FLÓREZ, R., (1999). Evaluación, pedagogía y cognición. McGraw-Hill Interamericana, S. A, Santafé de Bogotá, D.C Colombia.

GUIBERT, N., GUITTET, L. y GIRARD, P. (2005). A study of the efficiency of an alternative programming paradigm to teach the basics of programming. Disponible en: <http://www.lisi.ensma.fr/ftp/pub/documents/papers/2005/2005-WCCE-Guibert.pdf>. Recuperado el 20 de junio de 2014.

HEDERICH, C., y CAMARGO, A., (1999). Estilos cognitivos en Colombia. Resultados en cinco regiones culturales colombianas. Universidad pedagógica nacional-Colciencias, Santafé de Bogotá.

HEDERICH, C., (Diciembre, 2004). Estilo cognitivo en la dimensión independencia-dependencia de campo – influencias culturales e implicaciones para la educación. Tesis doctoral. Universidad autónoma de Barcelona. Recuperado el 20 de abril de 2014 desde <http://hdl.handle.net/10803/4754>

JUSTICIA, J. F. (1999). Análisis factorial de las escalas ACRA en una muestra de alumnos universitarios. Mente y conducta en situación educativa. Universidad de Valladolid. Revista electrónica del Departamento de Psicología, 51-66.

JHON, A. y GARZÓN, S. Sistema de personalización Web para el proceso de aprendizaje en una plataforma de educación.

MARTINEZ, Y. (2005). En busca de una nueva forma de enseñar a programar. Investigación bibliográfica. Disponible en: [http://www.mty.itesm.mx/rectoria/dda/rieee/pdf-05/27\(DTIE\).YolandaMtz..pdf](http://www.mty.itesm.mx/rectoria/dda/rieee/pdf-05/27(DTIE).YolandaMtz..pdf) Recuperado el 4 de abril de 2014.

Metodología de la programación. Tercera Edición. Editorial Alfaomega.

OVIEDO, M. y ORTIZ, F.G. (2002). La enseñanza de la programación. Disponible en: <http://bibliotecadigital.conevyt.org.mx/colecciones/documentos/somece2002/Grupo4/Oviedo.pdf>. Recuperado el 25 de mayo de 2014.

POLYA, G. (1965). "Como plantear y Resolver problemas", Trillas, 1965.

POSADA, A., R. (2005). Pensando la calidad de la educación. Conferencia presentada en el Congreso por una educación de calidad en el Caribe colombiano, Agosto, Barranquilla.

STALLMAN, R. Software libre para una sociedad libre. Traficante de sueños, 2004