

Ejercicios en computador vs. ejercicios en papel para enseñar a programar: un estudio comparativo¹

Omar Iván Trejos Buriticá²

Recibido, Junio 28 de 2017
Concepto evaluación, Septiembre 1 de 2017
Aceptado, Septiembre 1 de 2017

Referencia: Trejos Buriticá, O. I. (2018). "Ejercicios en computador vs. ejercicios en papel para enseñar a programar: un estudio comparativo". *Revista Academia y Virtualidad*, 11, (1), 1-15

Resumen

El presente artículo expone los resultados de una investigación realizada en el aula en la asignatura Programación I. Con la investigación se pretendió conocer el impacto de dos metodologías de resolución de problemas usando la lógica de programación y los lenguajes para su implementación. Se ha realizado desde una óptica cualitativa, en lo que se refiere al desempeño de los estudiantes como programadores, y cuantitativa, en lo que corresponde a la valoración de las evaluaciones escritas. Se acudió a la estructuración de dos cursos en paralelo, cada uno utilizando metodologías diferentes. Con un grupo se implementó una metodología basada en la resolución conceptual de dichos problemas a partir del planteamiento de soluciones en el papel y con el otro se ha acudido a la resolución de problemas usando el computador. Los resultados evidencian la necesidad de fomentar la lógica y el planteamiento de soluciones en el papel frente al uso del computador. Se hacen algunas conclusiones que dejan abiertas las puertas de la discusión acerca de los diferentes y posibles caminos que tengan los estudiantes de programación para apropiar, asimilar y aplicar los conocimientos propios de esta área temática.

Palabras clave: educación; enseñanza y formación; métodos de enseñanza; educación a distancia.

¹ Artículo de investigación científica y tecnológica

² PhD en Ciencias de la Educación. Profesor titular del programa de Ingeniería de Sistemas y Computación en la Facultad de Ingeniería de la Universidad Tecnológica de Pereira, Colombia. E-Mail: omartrejos@utp.edu.co.

Exercises with computer vs. Exercises in paper: a comparative study

Abstract

This article presents the results of a research in the classroom in the subject Programming I, initial course of computer programming in Systems Engineering and Computing. The research pretend to approach the impact of two methodologies of problem solving based on programming logic and the use of programming languages for its implementation. We use a qualitative perspective, as far as the students' performance as programmers, and quantitative, in what corresponds to the qualification of the written evaluations. It has been used to structure two parallel courses, using different methodologies for solving programming problems. With the first group, a methodology was implemented based on the conceptual resolution of these problems using solutions on paper and with the other one we use resolution of problems based on the intensive use of the computer. The results show a very interesting trend that allows us to make some important inferences regarding the promotion of logic and the approach of solutions in the paper against the use of the computer and the deepening in the purely technological knowledge. Conclusions leave open doors of the discussion about the different and possible ways that the students of programming to appropriate, assimilate and apply the own knowledge of this area.

Keywords: education; teaching and training; teaching methods; distance education.

Exercícios de computador vs. Exercícios em papel: um estudo comparativo

Resumo

Este artigo apresenta os resultados de pesquisa realizada na sala de aula no Anexo I. O sujeito da pesquisa foi concebido para aproximar o impacto dos dois métodos de resolução de problemas usando programação e linguagens lógica para a implementação. Foi realizada a partir de um ponto de vista qualitativo, em relação ao desempenho do aluno como programadores, e quantitativa no que se refere à avaliação de avaliações escritas. Ele foi para a estruturação de dois cursos em paralelo, cada um usando diferentes metodologias. Com um grupo de uma metodologia baseada na resolução conceitual de tais problemas de desenvolvimento de soluções em papel eo outro foi para resolver problemas usando o computador foi implementado. Os resultados mostram a necessidade de promover a lógica e desenvolvimento de soluções em papel contra o uso do computador. algumas conclusões que deixam portas discussão aberta sobre os diferentes caminhos possíveis que os alunos têm de se apropriar de programação, assimilar e aplicar o conhecimento desta área assunto são.

Palavras chave: educação ensino e formação; métodos de ensino; educação a distância.

Introducción

El problema a investigar se centra en la búsqueda de un camino más apropiado para fortalecer el aprendizaje, la asimilación y la aplicación de los conceptos que subyacen a la lógica de programación. Para ello, se acude a dos metodologías que el mismo quehacer docente ha ido estructurando: la resolución de enunciados de programación a partir del planteamiento de soluciones en papel, en primera instancia, y el planteamiento de soluciones a partir de la utilización intensiva del computador, directamente, en el proceso de concepción de las mismas. La pregunta de investigación gira alrededor de cuál de los dos caminos es el más expedito para que, en función del tiempo, los conceptos propios de la lógica de programación sean interiorizados por los estudiantes de programación en los primeros cursos de esta área en un programa de Ingeniería. Se ha considerado este como un problema de una importancia relevante debido a que en Colombia existen más de 200 programas de Ingeniería de Sistemas y los cursos básicos de programación de computadores se imparten no solo en estos sino en más de 700 programas de formación profesional tanto de ingeniería como de otras áreas (Ministerio de Educación Nacional, 2015). El propósito de este estudio es hacer un aporte a la discusión de si lo más importante, en las etapas infantiles del aprendizaje de la programación de computadores dentro de un contexto de formación profesional, es atender primero el fortalecimiento de la lógica para una posterior aplicación a partir de la tecnología computacional o comenzar por la utilización intensiva de computadores, amén de que corresponden al entorno natural de los jóvenes de hoy, para que estos, a partir de su uso, infieran los conceptos básicos y los apropien por un camino más expedito.

La enseñanza de la programación de computadores constituye un factor de gran importancia dentro del contexto de la formación tanto de ingenieros de sistemas como de ingenieros de otras áreas toda vez que, a partir de ella, se interactúa con conceptos de una lógica

que, sin ser la lógica deliberativa natural del ser humano, es la que se utiliza para aprovechar y capitalizar las bondades y ventajas de las nuevas tecnologías de la información y la comunicación.

El presente artículo es un producto del proyecto de investigación 6-16-13 titulado *Desarrollo de un modelo metodológico para el aprendizaje de la programación en Ingeniería de Sistemas basado en Aprendizaje Significativo, Aprendizaje por Descubrimiento y el Modelo 4Q de preferencias de pensamiento*, avalado por el Consejo de Facultad de Ingenierías y aprobado por la Vicerrectoría de Investigaciones, Extensión e Innovación de la Universidad Tecnológica de Pereira de Colombia. La temática abordada en esta investigación tiene una especial relevancia toda vez que los docentes ingenieros se han formado en el campo de la ingeniería mas no en el campo de la docencia y, por lo tanto, el abanico de formas diversas como imparten las asignaturas de programación son tan disímiles como, metodológicamente, diferentes.

Sin embargo, un sondeo realizado verbalmente durante el desarrollo de este proyecto de investigación entre los docentes del área de programación de seis universidades del centro de Colombia, revela que para ellos la enseñanza de la programación tiene un único objetivo: que los estudiantes aprendan a programar y, por lo tanto, la pregunta de investigación ha sido altamente coincidente con sus preocupaciones, pues son pocos los estudios, bajo el marco de una investigación, que develen las inferencias que se pueden derivar de la resolución de problemas usando computador en comparación con la resolución de los mismos problemas usando el papel y que tenga incidencia en el aprendizaje de la programación.

Todo lo anterior, teniendo en cuenta el contexto tecnológico de los jóvenes de hoy, pues, para ellos, la tecnología se ha constituido en parte de su paisaje propio así como de su lenguaje natural. Esta última aclaración se hace para que se conozcan las condiciones en las cuales se ha realizado

la presente investigación y de donde se han obtenido los resultados que aquí se presentan. Una interesante aplicación de los resultados de esta investigación radica en que la metodología utilizada se podría utilizar para otras áreas que tienen una connotación similar a la de la programación, ya que tiene una arista teórica pura que puede llevarse a la práctica a través de la utilización y mediación de algún tipo de instrumento. En este campo bien pueden incluirse las áreas de la física, la biología, la química como asignaturas de las ciencias básicas y la dinámica, la electrónica, la mecánica de fluidos, la electrostática, el cálculo de fuerzas y otras como asignaturas de las ciencias aplicadas como otras ingenierías.

Lo innovador de este artículo es que se presentan resultados e inferencias de dos cursos completos de programación de computadores, cada uno utilizando una metodología diferente para el planteamiento de las soluciones a problemas enunciados, y se ha aplicado todo el rigor de la metodología de la investigación científica de manera que los resultados puedan ser replicables y obtenibles en las mismas condiciones o en condiciones similares e, igualmente, en áreas similares. Con esto quiere decirse que se ha acudido a una metodología de investigación de las llamadas ciencias blandas para inferir conclusiones de la interacción con ciencias duras. Es de aclarar, de todas formas, que el presente estudio no establece una verdad sino solamente una aproximación a la realidad, es decir, que deja algunas reflexiones que pueden ser de gran importancia para aquellos docentes ingenieros que, en sus respectivas áreas temáticas, quieren cuestionar un poco su quehacer y buscar caminos más eficaces en el logro de sus objetivos de aprendizaje. Para el desarrollo de esta investigación se ha acudido a la teoría de aprendizaje significativo, a la teoría de aprendizaje por descubrimiento, a la teoría del conectivismo y al *active learning*, que se explicarán un poco más adelante. La investigación se realizó con dos grupos (en paralelo) de la asignatura Programación I de Ingeniería de Sistemas y

Computación de la Universidad Tecnológica de Pereira durante los semestres I-2015, II-2015, I-2016, II-2016 y I-2017. Los datos correspondientes a los perfiles, interacciones, evaluaciones y resultados tanto cuantitativos como cualitativos se presentan en numerales posteriores.

De acuerdo a lo expuesto, ¿es posible concluir, en relación con el aprendizaje de la programación de computadores, qué metodología es más favorable entre dos utilizadas para este propósito? Esa es la hipótesis que se plantea en consonancia con el objetivo de la presente investigación y, si bien, no necesariamente se pueda encontrar una respuesta absolutamente paramétrica que satisfaga la necesidad presentada, sí es posible al menos brindar elementos de juicio suficientemente sólidos que aporten a la discusión y que muestren resultados comparativos que solidifiquen la opinión y experiencia de los docentes.

El artículo está organizado en formato IMRYD a partir del cual se expone una introducción que se complementa con un marco teórico. Posteriormente se presenta la metodología utilizada en detalle y seguidamente se muestran los resultados obtenidos para hacer sobre ellos una discusión y, finalmente, aportar unas conclusiones que son las que, se espera, enriquezcan los debates que al respecto se hagan en las salas de profesores de los departamentos de programación de computadores de las diferentes universidades.

Conceptos fundamentales

El aprendizaje significativo se ha constituido en uno de los pilares más importantes del aprendizaje moderno, toda vez que el conocimiento adquiere significado, lo cual lo posiciona en un lugar privilegiado en donde se enlazan la teoría (que se adquiere en las aulas de clase dentro del marco de los programas de formación profesional) y la práctica (que se ejecuta en la vida real y que cristaliza reglas y excepciones aprendidas desde la teoría) (Ausubel, 1963). Según Ausubel

(1986), lo más importante en un proceso de aprendizaje es lo que el estudiante ya sabe, y esta teoría parte de que todo conocimiento consciente siempre tiene un conocimiento previo asociado que es lo que hace que se puedan modificar las estructuras cognitivas para mejorar, actualizar o reemplazar lo que ya se sabe.

Desde el entorno externo hasta el mismo salón de clases, todos ellos se configuran en contextos desde los cuales los estudiantes pueden aprender y que, de una u otra forma, establecen las bases para que se puedan abordar nuevos conocimientos (Barriga Arceo & Hernández Rojas, 2002). De esta forma, el conocimiento se basa en unos conocimientos previos, en un nuevo conocimiento y en una capacidad de establecer relaciones de sustitución, actualización o descarte entre dichos conocimientos previos y los nuevos.

De igual manera, se constituye en un factor de gran importancia para capitalizar un proceso de aprendizaje la actitud del estudiante, lo que se fundamenta básicamente en la motivación que tenga hacia el objeto de aprendizaje (Bonwell & Eison, 1991) y en la capacidad que, con colaboración del docente y el seguimiento a las actividades, pueda desarrollar para establecer nexos entre el conocimiento previo y el nuevo conocimiento. La motivación es el ánimo positivo que tenga el estudiante hacia lo que quiere aprender (Brown Wright, 2011) y determina la facilidad con que se acceda, especialmente, al nuevo conocimiento, ya que el conocimiento previo, en lo normal, siempre lo tiene.

Por su parte, el aprendizaje por descubrimiento establece que el ser humano aprende mucho más fácil todo aquello que descubre que aquello que se le enseña completamente (Bruner, 1963). Esto significa que cuando se trata de asignaturas que tienen tanto un componente teórico bien fundamentado como un componente práctico mediado por herramientas (computadores, para el caso actual) es posible brindarle al estudiante los mecanismos para que “descubra” el conocimiento, todo ello dentro de un marco coherente

de premios y castigos (Bruner, 2009), de forma que los retos puedan ser alcanzables, lo cual no significa que sean “fácilmente alcanzables” por todos los estudiantes.

El aprendizaje por descubrimiento tiene una relación directa con la enseñanza y el aprendizaje de la programación de computadores (Blanco Rivero y Silva Sánchez, 2009) dado que para resolver problemas computacionales debe acudir a una lógica que no corresponde a la lógica humana natural y deliberativa sino que corresponde a una lógica binaria paramétrica cuyos elementos constitutivos son significativamente más simples que los que se utilizan en los razonamientos propios del ser humano. Cada enunciado lógico que se pueda resolver y que, posteriormente o antes, se pueda implementar en el computador constituye un reto alcanzable, y alcanzado, por el estudiante, así como cada enunciado que no se haya resuelto corresponde a una dificultad que el alumno, con el acompañamiento efectivo del docente, debe superar interiorizando la teoría de la programación que subyace a su aplicación en el computador o practicando con tal destreza el uso del computador que pueda inferir apropiadamente los conceptos que correspondan.

Como Active Learning se conoce un método de enseñanza y aprendizaje orientado a la acción, en el cual el estudiante, con la ayuda del docente, se compromete a establecer sus propias metas, a estudiar para alcanzarlas, a evaluarlas y a establecer los correctivos que sean necesarias en su camino a lograr sus objetivos de aprendizaje (Prince, 2004). El Aprendizaje Activo implica una alta capacidad del docente para lograr que sus estudiantes alcancen, por sí solos, las metas que se han propuesto dentro lo esperado en sus respectivos procesos de aprendizaje. En este método, el estudiante es consciente del compromiso de aprender, de la necesidad de aprender y de la utilidad de aprender (Prieto, 2006).

En cuanto al compromiso, el estudiante deberá asumir un rol altamente participativo de manera que se despierte en él la motivación por el objeto que se quiere aprender. En cuanto

a la necesidad de aprender, el alumno deberá conocer, bien por sus propios medios o de manera guiada por el docente, las implicaciones tanto verticales como horizontales de adquirir (o de no adquirir) determinado conjunto de conocimientos (Jones, 2007). En cuanto a la utilidad de aprender, el estudiante deberá encontrar espacios de aplicación en los cuales el conocimiento se haga efectivo por su mediación e interacción con el mundo real, bien sea dentro del contexto puramente académico o dentro del contexto de su vida cotidiana. Esto tiene una alta relación con el concepto de significado que se expuso anteriormente en el aprendizaje significativo. El método conocido como Active Learning constituye un excelente complemento para que los procesos de aprendizaje, con la participación y compromiso de los estudiantes, se conviertan en aventuras positivas del conocimiento de donde siempre se obtengan resultados favorables (Azad & Smith, 2014).

Por su parte, el conectivismo corresponde a una teoría de aprendizaje que se ha planteado recientemente y que se cristaliza en la actual era digital (Attard, Di Ioio & Geven, 2010). Fue desarrollada por George Siemens y Stephen Downes y se basa en un análisis del efecto que la tecnología tiene en la forma en que vivimos en tiempos modernos, en la manera como nos comunicamos y, especialmente, en la forma como aprendemos. En el conectivismo, por primera vez, se acepta que el conocimiento podría no estar en el ser humano sino en alguna forma de almacenamiento (Trejos Buriticá, 2013) y que, por tanto, deberá desarrollarse en el estudiante una gran habilidad para encontrar la información que requiere para fortalecer un determinado conocimiento académico.

Según sus autores, el aprendizaje se produce a través de las conexiones dentro de las redes y para ello se concibe un modelo que acude a una red de nodos y nexos (enlaces) que definen el aprendizaje. Para ello, se define el aprendizaje como un conjunto de procesos que suceden en una gama bastante amplia de ambientes que no siempre están controlados por el individuo que

quiere aprender (Calvani, 2008). De esta forma, se acepta que el aprendizaje resida (o por lo menos pueda residir) fuera del ser humano, bien en una organización de datos, bien en una expresión tecnológica, en una base de datos o en un contexto al cual aún no se haya accedido. Esta teoría se enfoca en las diferentes conexiones que se pueden establecer, de manera especializada, con conjuntos de información que permiten aumentar el nivel actual del conocimiento (Kennedy, 2009).

Se deja claro entonces que, a la luz de esta teoría, las decisiones se basan en una transformación acelerada de las bases de conocimiento y que se convierte en un ciclo bastante breve la permanencia activa de determinada información así como su obsolescencia. Esto implica que permanentemente el estudiante tiene el reto de aprender a distinguir entre la información verdaderamente importante y la que podría llamarse como trivial. Igualmente, tal como lo plantea la teoría del aprendizaje significativo (Guerrero Sánchez, 2014), por la interacción con los medios electrónicos modernos, el alumno debe desarrollar una gran capacidad para poder reconocer los elementos de juicio que realmente cambian o modifican de manera sustancial su conocimiento.

La enseñanza de la lógica de programación siempre se ha convertido en un factor de gran importancia de cara a los objetivos de aprendizaje de un curso de programación (Trejos Buriticá, 2000). Como lógica de programación se conoce el conjunto de fundamentos teórico-matemáticos que subyacen a la programación, independiente del paradigma que se quisiera abordar (Trejos Buriticá, 2006). Tradicionalmente el aprendizaje de la lógica de programación ha sido el tema antecesor de la programación como tal (Van Roy, 2008), pues la programación es la cristalización de las soluciones planteadas sobre la base matemática y llevadas al computador a través de un compilador y usando un lenguaje de programación formal (Schildt, 2010). Precisamente, lo que se pretende con esta investigación es aportar a la discusión, y en el mejor de los casos

brindar una propuesta de solución, al dilema de qué se debe fortalecer primero en los estudiantes de programación con las características de los estudiantes de hoy: la lógica de programación (con todos sus fundamentos y teorías) o la programación (con toda la tecnología disponible para hacerlo) y, con ello, poder determinar cuál de los dos caminos provee mayor solidez al conocimiento necesario para que el estudiante aprenda a programar.

Metodología

La investigación se realizó desde una perspectiva cualitativa para conocer, por opinión de los mismos estudiantes, el avance y apropiación de los conceptos que subyacen a la lógica de programación por dos caminos diferentes y desde una perspectiva cuantitativa intentando cuantificar lo aprendido y, de esa forma, poder comparar los resultados de los dos grupos. Para ello, se trabajó con dos grupos diferentes. En el primero, se impartieron las clases y se plantearon enunciados para que fueran resueltos en el aula durante las sesiones normales sin la intervención de ningún tipo de elemento tecnológico. En el segundo, se impartieron las clases y se plantearon enunciados, desde el principio, con la utilización intensiva del computador y los recursos que proporciona para el desarrollo del curso de programación. La tabla 1

expone los grupos que se atendieron durante la investigación.

Es de aclarar que la asignatura Programación I en Ingeniería de Sistemas y Computación de la Universidad Tecnológica de Pereira se basa en el desarrollo, apropiación y aplicación del paradigma funcional a partir del uso de DrRacket como herramienta computacional. De la misma forma, se optó por realizar este experimento con los estudiantes de primer semestre, pues se considera que de la apropiación de los conceptos básicos que subyacen a la programación depende el desarrollo y apropiación de los contenidos de las asignaturas que siguen en esta línea de profundización.

La asignatura se organizó con base en la experiencia del investigador como docente de programación y como programador, intentando que cada tema cubriera tanto en lo teórico como en lo práctico una semana de clases. De esta forma se estructuró la asignatura a partir de 16 temas que correspondían a las 16 semanas de clases. Se buscó igualmente que la secuencia de temas tuviera una lógica escalonada, es decir, que cada tema recogiera el anterior y de esta forma se fuera avanzando progresivamente en el fortalecimiento de los temas anteriores y en la adquisición de nuevos conocimientos.

Igualmente, se logró que la organización temática por semanas de la asignatura

Tabla 1.

Grupos atendidos durante la investigación

Año	Semestre	Grupo	Característica	No. de estudiantes
2015	I	01	Trabajo en el aula	19
		02	Trabajo en la sala de computadores	21
	II	01	Trabajo en el aula	20
		02	Trabajo en la sala de computadores	21
2016	I	01	Trabajo en el aula	18
		02	Trabajo en la sala de computadores	17
	II	01	Trabajo en el aula	19
		02	Trabajo en la sala de computadores	22
2017	I	01	Trabajo en el aula	19
		02	Trabajo en la sala de computadores	18

Fuente: elaboración propia.

estuviera dividida en unidades de contenido completas, esto es, las primeras cuatro semanas correspondían a la apropiación de los conceptos básicos del paradigma funcional como son el concepto de función y el paso de argumentos; las siguientes cuatro, se destinaron al estudio de los condicionales; las cuatro que seguían se destinaron al estudio de la recursividad, y las últimas cuatro, se dedicaron a la definición, utilización y manejo de conjuntos de datos. De esta forma se logró que, mentalmente, los estudiantes pudieran tener un plano del contenido de la asignatura y que cada parte de dicho plano estuviera plenamente definida en su contenido y en sus límites.

Para los efectos evaluativos de la investigación, la asignatura Programación I se dividió en tres parciales y un examen final, cada uno con un peso porcentual de 25% sobre la nota definitiva. Así mismo, el contenido temático de la asignatura se organizó de forma que se pudieran realizar las evaluaciones sobre temas completos cada cuatro semanas. La tabla 2 presenta esta información de una manera más detallada.

Para el cálculo del promedio de las notas parciales y la nota final, en cada evaluación se excluyeron las cuatro mejores notas y las cuatro peores, basados en el hecho de que los valores extremos afectan los promedios y podrían afectar las inferencias que se hagan a partir de los resultados obtenidos. La asignatura cuenta con tres sesiones semanales, cada una de dos horas, para el desarrollo de los contenidos. Cada ítem del contenido correspondió a una semana específica de clases. De las tres sesiones semanales, se destinó la primera sesión para exponer, desde una perspectiva magistral, todo el corpus conceptual que implicaba el contenido en la semana correspondiente.

La segunda sesión se destinó para realizar ejercicios y resolverlos entre el docente y los estudiantes y la tercera, que fue la que fortaleció la investigación, se destinó para que los estudiantes, con el acompañamiento del docente pero cuidando que fueran ellos quienes encontraran las soluciones, resolvieran enunciados que se les entregaban al inicio de la sesión y que siempre fueran de diez puntos. En esta sesión se realizaba el

Tabla 2.
Organización de la asignatura Programación I

Semestre Inicial	Semestre Final	Evaluación	Contenido	%
1°	4°	Parcial 1	<ul style="list-style-type: none"> • Introducción • Resolución de problemas • El concepto de función • Llamados entre funciones 	25%
5°	8°	Parcial 2	<ul style="list-style-type: none"> • El concepto de toma de decisiones • Condicionales simples • Condicionales compuestas • El concepto de menú 	25%
9°	12°	Parcial 3	<ul style="list-style-type: none"> • El concepto de recursividad • Recursividad 1° nivel • Recursividad 2° nivel • Recursividad 3° nivel 	25%
13°	16°	Ex. Final	<ul style="list-style-type: none"> • El concepto de conjuntos de datos • Listas • Vectores • Gráficos y GUI 	25%
				100%

Fuente: elaboración propia.

trabajo en el aula (para los grupos identificados como 01) y en la sala de computadores (para los grupos identificados como 02 en la tabla 1). De esta última sesión fue de donde se obtuvieron los resultados que se exponen en este artículo. Las evaluaciones se hicieron en ambos grupos a la misma hora y con un contenido exactamente igual para poder establecer un comparativo entre los dos grupos. Vale la pena tener en cuenta que para la resolución de los parciales, los estudiantes de los grupos 01 contaron con un tiempo de dos horas y con papel y lápiz para resolver los enunciados. Por su parte, los grupos 02 tuvieron el mismo tiempo disponible y contaron con un computador con el IDE DrRacket para resolver directamente los enunciados con el computador. Es de anotar que todos los estudiantes, por fuera de las sesiones, tenían acceso libremente a los recursos y servicios que provee Internet y, por ello, todo este proceso está mediado por la teoría del conectivismo, pues en tiempos modernos limitar el acceso a las nuevas tecnologías por fuera del aula (excepto que sea una política estatal) es sencillamente imposible. Debe acotarse que el acceso a estas nuevas formas de comunicación es un posible camino para que el estudiante, motivado, mejore sus niveles de aprendizaje y la

apropiación de los conceptos que subyacen al paradigma funcional.

Resultados

La valoración promedio de las evaluaciones parciales y del examen final se presenta en la tabla 3 por cada uno de los cursos analizados. Debe recordarse que los grupos numerados como 01 son los que resolvieron sus enunciados usando solo papel y lápiz y los grupos numerados como 02 son los que se apoyaron directamente en el computador y en sus herramientas de programación.

El concepto emanado por parte de los estudiantes, en relación con el proceso investigativo, se muestra en la tabla 4, en la cual se han seleccionado (al azar) algunas de esas opiniones para que se pueda dar fe de la parte puramente cualitativa. Con el ánimo de que el estudiante tuviera un espacio libre para opinar, se planteó un instrumento con preguntas abiertas de manera que se pudieran obtener opiniones con un tono más personal y, para efectos de esta investigación, más confiable.

Dicho instrumento contenía tres preguntas, no era necesario que las respondieran todas, las podían responder anónimamente (opción a la cual el 95% de los estudiantes se acogió) y todo tipo de opinión, tanto en el

Tabla 3.

Promedio de las notas obtenidas en las evaluaciones

Año	Semestre	Grupo	No. de estudiantes	Estudiantes Evaluados	P1	P2	P3	EF	Prom.
2015	I	01	19	11	3.7	3.8	3.7	3.9	3.775
		02	21	13	3.4	3.6	3.6	3.8	3.600
	II	01	20	12	3.9	3.9	4.0	4.1	3.975
		02	21	13	3.8	3.9	3.9	4.0	3.900
2016	I	01	18	10	3.5	3.6	3.5	3.4	3.500
		02	17	9	3.5	3.6	3.4	3.2	3.400
	II	01	19	11	3.6	3.6	3.7	3.6	3.625
		02	22	14	3.5	3.6	3.5	3.5	3.525
2017	I	01	19	11	3.7	3.8	3.9	4.0	3.850
		02	18	10	3.5	3.5	3.6	3.6	3.550

Fuente: elaboración propia.

Tabla 4.
Concepto personal del proceso

Año	Semestre	Grupo	No. de estudiantes	Estudiantes evaluados	Algunas opiniones
2015	I	01	19	11	<ul style="list-style-type: none"> • Qué bueno liberarse un poco de la tecnología • Debiera usarse el computador • Estoy aprendiendo a programar
		02	21	13	<ul style="list-style-type: none"> • Creo que esto sí es programación • ¿Y el que no tiene computador qué? • Muy fácil programar con el computador
	II	01	20	12	<ul style="list-style-type: none"> • Falta ver si esto sí funciona en el computador • Me parece muy bueno programar así • No parece que estuviéramos programando
		02	21	13	<ul style="list-style-type: none"> • No veo el avance • Creo que estoy avanzando bastante • Ya sé programar
2016	I	01	18	10	<ul style="list-style-type: none"> • Me siento un ingeniero • Definitivamente programar es muy fácil • Voy por la de oro
		02	17	9	<ul style="list-style-type: none"> • Este computador a veces se traba • No siempre funciona como uno cree • Me parece muy fácil programar así
	II	01	19	11	<ul style="list-style-type: none"> • Creo que esto es ser ingeniero • Me siento un ingeniero programador • Me rinde bastante resolver problemas así
		02	22	14	<ul style="list-style-type: none"> • ¿Otro problema? • Me siento avanzando bastante • Sin el computador no sé qué haría
2017	I	01	19	11	<ul style="list-style-type: none"> • Creo que no voy por buen camino • Veo el avance en cada ejercicio • ¿Funcionarán estas soluciones?
		02	18	10	<ul style="list-style-type: none"> • Me conozco este compilador completo • Sé usar este entorno de programación • ¿Cómo será programar sin el computador?

Fuente: elaboración propia.

fondo como en la forma, era absolutamente válida y aceptada. Igualmente, las respuestas podían ser tan extensas o breves como el estudiante quisiera. Las preguntas del instrumento fueron: a) ¿Cómo se sintió durante el desarrollo de la asignatura dentro del marco metodológico que se adoptó? b) ¿Qué aprendió durante el proceso de aprendizaje de la asignatura? c) ¿Qué críticas, aportes, sugerencias, observaciones, mejoras o elogios le haría al proceso de aprendizaje? Es de anotar que los estudiantes desde el principio fueron informados al respecto de todo el proceso,

tanto en su parte metodológica como en su parte temática y pedagógica.

El resumen de este cuadro, en relación con un criterio acerca de que la opinión de los estudiantes sea favorable o desfavorable al proceso, se presenta en la tabla 5.

Considerando que son opiniones favorables aquellas que resaltan las bondades de la metodología y que destacan los resultados obtenidos de manera individual por parte de cada estudiante y que son opiniones desfavorables las que pueden ir en un sentido contrario, sin desconocer que son opiniones enriquecedoras para el estudio, se realizó la

Tabla 5.
Resumen de opiniones

Año	Semestre	Grupo	No. de estudiantes	Estudiantes evaluados	Opiniones favorables	Opiniones desfavorables
2015	I	01	19	11	8	3
		02	21	13	10	3
	II	01	20	12	11	1
		02	21	13	11	2
2016	I	01	18	10	8	2
		02	17	9	9	0
	II	01	19	11	11	0
		02	22	14	12	2
2017	I	01	19	11	10	1
		02	18	10	10	1

Fuente: elaboración propia.

clasificación de opiniones que se muestra en la tabla 5.

La estadística de mortandad de los estudiantes en el curso de Programación I se presenta en la tabla 6.

Vale la pena tener en cuenta que algunos de los estudiantes que no aprobaron el curso de programación tuvieron razones de índole puramente personal para no haberlo logrado. Entre estas razones se cuentan: el traslado a otra ciudad, el cambio de carrera, fallecimiento de la persona que lo patrocinaba, problemas personales, embarazo inesperado, entre otras. Aunque no se hizo un análisis detallado de las razones, se po-

dría estimar que la mitad de los estudiantes que no aprobaron el curso Programación I deben esta situación a razones enteramente académicas: falta de preparación, falta de estudio, falta de motivación o falta de interés por aprender los conceptos que conforman el corpus del contenido de la asignatura.

Discusión

Lo primero que vale la pena analizar es la metodología utilizada. Durante muchos años de experiencia, vivida por el autor de este artículo y referida por muchos colegas, la gran preocupación en la enseñanza de la

Tabla 6.
Estadística de mortandad al final del curso

Año	Semestre	Grupo	No. de estudiantes	Pasaron	No pasaron
2015	I	01	19	19	0
		02	21	19	2
	II	01	20	19	1
		02	21	18	3
2016	I	01	18	18	0
		02	17	15	2
	II	01	19	18	1
		02	22	20	2
2017	I	01	19	18	1
		02	18	16	2

Fuente: elaboración propia.

programación de computadores ha radicado en cuál debe ser el camino más fácil para que los estudiantes aprendan a programar, es decir, para que cuando se enfrenten a una situación problemática que puede ser resuelta con el computador puedan encontrar la solución a partir de los conceptos y teorías que han aprendido y que posibilitan una solución paramétrica, todo ello a partir de la iniciativa y creatividad propia del estudiante.

En este proyecto de investigación se buscó establecer unas condiciones idénticas para que, en paralelo, se pudieran analizar dos grupos dentro del marco del proceso de aprendizaje de la programación en su fase inicial frente al programa de estudios de Ingeniería de Sistemas y Computación. Se procuró que todas las condiciones fueran las mismas para tener una perspectiva lo más objetiva posible. Los horarios fueron los mismos en los dos grupos, los salones para la exposición magistral fueron muy similares, al punto que estaban ubicados en el mismo edificio, en el mismo piso y con las mismas condiciones de luz, exceptuando la tercera sesión semanal (sesión de ejercicios en clase), que para un grupo se realizaba en el aula y para el otro se realizaba en la sala de computadores. Es de anotar que, de todas formas, los estudiantes habrían accedido a estas formas de acceso al conocimiento, sea que se les avalare o no. Se hicieron sesiones ocupando el mismo tiempo y explicando los mismos temas de acuerdo a la secuencia temática que se expone en la tabla 2.

Todo esto con el ánimo de obtener unos resultados con alto nivel de confiabilidad en relación con las condiciones de desarrollo de la investigación pues así lo exigía el trabajo con los dos grupos en paralelo. Se hicieron todos los esfuerzos para que todas las condiciones fueran similares, pero eventualmente pudieron haber existido diferencias mínimas que, si bien para el investigador es posible que fueran imperceptibles, para los estudiantes es posible que fueran significativas, como por ejemplo en una oportunidad un concierto que se estaba desarrollando en la parte externa del bloque universitario

en donde se impartían las clases, pues se escuchaba notoriamente más fuerte en un salón que en otro, mientras se estaba desarrollando una evaluación. De allí por qué se quiso realizar este proceso investigativo con varios cursos en diferentes semestres.

En relación con el promedio de las notas, se puede observar, según lo que se presenta en la tabla 3, que los resultados son favorables en ambas formas metodológicas en escala de 1 a 5, teniendo en cuenta que de cada grupo se excluyeron las 4 notas más altas y las 4 notas más bajas para que los promedios no se afectaran por los valores extremos. En este sentido es posible que se pudiera pensar, desde el plano estadístico, que usar como medida de tendencia central la desviación estándar pudiera haber sido más apropiado. Sin embargo, y para este caso investigativo específico, el promedio (como dato) podía decir muchas más cosas en relación con la realidad a la cual se quiere aproximar este estudio.

Se puede notar que, aunque los promedios son favorables, en todo momento se puede percibir que los estudiantes que resolvieron sus enunciados y evaluaciones con el computador, siempre obtuvieron notas inferiores a aquellos que hicieron lo mismo pero en el aula convencional. Sin embargo, puede inferirse que la búsqueda y hallazgo de las soluciones lógicas para unos enunciados de programación, podrían estar más al alcance de la mano de aquellos estudiantes que, al resolverlos en el papel, deben acudir a los recursos que la matemática provee.

Es de anotar también que la diferencia entre los promedios obtenidos en los grupos que resolvieron talleres, ejercicios y evaluaciones en el aula convencional y los promedios de los grupos que lo hicieron en la sala de computadores es significativa, cuantitativa y cualitativamente hablando, pues siempre son diferencias que oscilan entre 0,1 y 0,5. Vale la pena tener en cuenta que, en algunos casos muy esporádicos, el promedio calculado fue igual en ambos casos. Tal vez esto indique que si bien el fortalecimiento de los conceptos matemáticos, como recurso primario y

excelso para un programador, corresponde a una estrategia que además de ser aparentemente acertada ha sido la forma tradicional de enseñar la programación, es muy posible que a partir de la interacción con el computador y con sus herramientas para buscar y encontrar soluciones a enunciados de programación, también se pueda llegar a una apropiación de los elementos básicos que subyacen a la teoría de la programación desde una perspectiva mucho más práctica y en términos del lenguaje natural de los jóvenes de hoy como es la tecnología.

Esta idea anterior puede discrepar de muchos autores que consideran que la enseñanza de la programación va directamente relacionada con el uso del computador y, precisamente, fue eso lo que esta investigación demostró: que si lo que se quiere es que los estudiantes apropien la lógica de programación el camino parece ser diferente a si lo que se quiere es que los estudiantes aprendan un lenguaje de programación.

Dado que todas las notas evaluativas tenían el mismo peso porcentual, el promedio de dichas notas es un valor que definía la nota definitiva, y el promedio de estos promedios corresponde a un dato que muestra que aunque la diferencia no supera el valor 0,300, de todas formas siempre es más alto en los estudiantes que trabajaron en el aula frente a los que trabajaron en la sala de computadores. En referencia a la opinión de los estudiantes, se pueden hacer algunas consideraciones que podrían enriquecer esta discusión. En general, y a partir del diálogo con ellos, todos se sintieron bastante cómodos con la metodología que le correspondió al grupo en el cual se encontraban, tanto los que tuvieron que fortalecer la lógica pura con los elementos que la matemática les proveía como los que tuvieron el acceso directo al computador y a sus herramientas de programación, opinaron de manera favorable en gran mayoría, lo cual coincide con el propósito de alcanzar un buen nivel de motivación de los alumnos por parte del profesor en relación con la asignatura en mención.

Posiblemente, por ser el computador un elemento moderno propio del lenguaje natural de los jóvenes de hoy, al reagrupar las opiniones favorables y desfavorables teniendo en cuenta la metodología usada en cada grupo, se observó que son menos las opiniones desfavorables en aquellos estudiantes que interactuaron con el computador que aquellos que resolvieron sus enunciados en el aula convencional, al punto de que hubo grupos (al menos 1) en el cual no hubo ninguna opinión desfavorable. Las opiniones textuales de los estudiantes develan, ante todo, una gran motivación por el contenido de la asignatura y una voluntad de querer estar en el proceso de aprendizaje independiente de la metodología que se utilizare.

Al hacer un balance de las notas definitivas de cada curso se observa, según lo muestra la tabla 6, que la mortandad entre los estudiantes que acudieron a las matemáticas como base para la lógica que posibilitara la solución de los enunciados fue notoriamente menor en comparación con la mortandad de los estudiantes que utilizaron el computador en el planteamiento e implementación de dichas soluciones. Claro está que no pueden descartarse razones de índole personal que no tuvieron relación con el proceso de aprendizaje, pero tampoco se puede desconocer que alguna incidencia tuvo el proceso de investigación implementado, pues, finalmente, y solo en primera instancia, pareciera ser más exitoso el proceso que se desarrolló en el aula que el proceso que se desarrolló en las salas de computadores en relación con la apropiación, asimilación y aplicación de los conceptos y teorías básicas que subyacen a la programación. De todas formas, este aparte del análisis de la mortandad en un curso de programación requiere un poco más de refinamiento en la depuración de la información para poder llegar a inferencias más objetivas.

Conclusiones

En relación con la pregunta de investigación puede decirse que efectivamente estos

dos caminos existen como estrategias de enseñanza de la programación y que ambos propenden por alcanzar la meta de que el estudiante aprenda a programar. Sin embargo, debe tenerse en cuenta, con los resultados, que si a los estudiantes de programación se les proporcionan los conceptos que les permitan acudir a las matemáticas como base para la solución de problemas, esto logrará en ellos una mentalidad mucho más amplia en el hallazgo y planteamiento de dichas soluciones y los liberará del condicionamiento que, por razones naturales, implica el uso intensivo de la tecnología en estas áreas temáticas.

Por su parte, la utilización del computador y de sus herramientas de programación asociadas podría ser un nuevo camino para que los estudiantes lleguen a concluir las bases de la programación a partir de su aplicación. Este artículo pretende aportar elementos de juicio a la discusión acerca de la enseñanza de la programación y sus verdaderos efectos en el proceso de aprendizaje por parte de los estudiantes. Es de anotar que vale la pena investigar, desde otras ópticas, los efectos de diferentes metodologías en relación con la apropiación y asimilación de los conceptos y teorías que subyacen a los fundamentos y a la lógica de la programación.

Conviene tener en cuenta que es posible que sean estos los tiempos en los cuales se deben cuestionar los procesos de formación tecnológica, a nivel de la programación de computadores dentro del marco de la formación de ingenieros de sistemas, y pensar no solo en las metodologías que tradicionalmente se han utilizado sino propender por otros caminos que, en lenguaje de los estudiantes de los tiempos modernos, logren los objetivos de aprendizaje en un tiempo menor amén de los cambios que permanentemente tienen las expresiones tecnológicas del mundo de hoy.

Referencias

- Attard, A., Di Ioio, E. & Geven, K. (2010). *Student Centered Learning. An insight into theory and practice*. Bucarest: Lifelong learning programme - European Community.
- Ausubel, D. (1963). *Psychology of meaningful verbal learning: an introduction to school learning*. New York: Grune & Straton.
- Ausubel, D. (1986). *Sicología educativa: un punto de vista cognoscitivo*. Ciudad de México: Trillas.
- Azad, A., & Smith, D. (2014). Teaching an introductory programming language in a general education course. *Journal of Information Technology Education: Innovations in Practice*, 13, 57-67.
- Barriga Arceo, F. y Hernández Rojas, G. (2002). *Estrategias docentes para un aprendizaje significativo: una interpretación constructivista*. Ciudad de México: McGraw Hill Interamericana.
- Blanco Rivero, L. y Silva Sánchez, E. (2009). *Herramientas pedagógicas para el profesor de Ingeniería*. Bogotá: Lemoine Editores.
- Bonwell, C., & Eison, J. (1991). Active Learning: Creating Excitement in the Classroom. *ASHEERIC Higher Education Report*, 1(1), 2-12.
- Brown Wright, G. (2011). Student centered learning in Higher Education. *International Journal of Teaching and Learning in Higher Education*, 23(3), 92-97.
- Bruner, J. S. (1963). *El proceso de la Educación*. Ciudad de México: Editorial Hispanoamericana.
- Bruner, J. S. (2009). *Actos de significado: más allá de la revolución cognitiva*. Madrid: Alianza Editorial.
- Calvani, A. (01 de 02 de 2008). Connectivism: new paradigm or fascinating potpourri. *Journal of e-Learning and Knowledge Society*, 247-252.
- Guerrero Sánchez, M. (2014). *Metodologías activas y aprendizaje por descubrimiento: las TIC y la educación*. Boston, MA: Amazon Digital Services.
- Jones, L. (2007). *Student Centered Learning*. Cambridge: Cambridge University Press.

- Kennedy, C. (30 de 11 de 2009). Connectivism at the technology age. B. S. University (Ed.). *Dr Rice EdTech 405*, 1-11.
- Ministerio de Educación Nacional. (2015). *Sistema Universitario Estatal SUE - Informe Estadístico Consolidado*. Bogotá: MEN, Presidencia de la República.
- Prieto, L. (2006). Aprendizaje Activo en el aula universitaria: el caso del aprendizaje basado en problemas. *Miscelánea Comillas. Revista de Ciencias Humanas y Sociales*, 64(124), 173-196.
- Prince, M. (2004). Does Active Learning work? *Journal Engineering Education*, 93(3).
- Schildt, H. (2010). *C Programming*. México: McGraw Hill.
- Trejos Buriticá, O. (2000). *La Esencia de la Lógica de Programación*. Pereira: Papiro.
- _____. (2013). *Significado y Competencias*. Pereira: Papiro.
- _____. (2006). *Fundamentos de Programación*. Pereira: Papiro.
- Van Roy, P. (2008). *Techniques and methods in programming computer*. Louvaine: University Press.