

Episteme vs. Tekné: una aproximación al aprendizaje de la programación de computadores en Ingeniería de Sistemas¹

Episteme vs. Tekné: an approach to the learning of computer programming in Systems Engineering

O. I. Trejos

Recibido: julio 27 de 2018 - Aceptado: mayo 30 de 2019

Resumen— El presente artículo expone los resultados de una investigación realizada en el aula, cuyo propósito fue el de establecer elementos de juicio para comparar dos estrategias de enseñanza y su impacto en el aprendizaje de la programación de computadores. Las dos estrategias comparadas fueron la enseñanza de la lógica de programación (la episteme), o la enseñanza de la aplicación de esta, reflejada en un lenguaje de programación (la tekne). Para medir su impacto, se segmentaron los cursos de Programación I en dos subgrupos, con el fin de poder realizar un seguimiento comparativo. El proceso se llevó a cabo con el 1er semestre de Ingeniería de Sistemas de una universidad pública. Los resultados se midieron desde lo cuantitativo con evaluaciones escritas idénticas y en condiciones similares para su resolución. También se realizó una aproximación cualitativa a la opinión de los estudiantes, al final del curso. Se concluye que pareciera ser más relevante la enseñanza de la episteme de la programación que de su aplicación, puesto que el estudiante puede ir de aquella a este. Sin embargo, se encontraron casos en los que algunos estudiantes, desde el estudio del lenguaje de programación, infirieron los conceptos subyacentes de la lógica pertinente.

Palabras clave— Aprendizaje, computadores, episteme, programación, tekne.

Abstract— This article exposes the results of a research made in the classroom looking for establishing elements of

comparative judgment that would allow comparing two teaching strategies, and their impact on learning, of computer programming in a Systems Engineering program. The two strategies to be compared consist of the teaching of the logic of programming (the episteme) or the teaching of its application reflected in a programming language (tekne). In order to measure the impact of the two strategies on the learning of programming, Programming I courses were divided into two subgroups so that a comparative follow-up could be carried out permanently. The process was carried out with the 1st semester of Systems Engineering of a public university. The results were measured quantitatively from identical written evaluations and under highly similar conditions for their resolution. A qualitative approach was also made based on the opinion of the students at the end of the course. It is concluded that the teaching of the episteme of programming seems to be much more relevant than its application since the student can go from the 1st to the other. However, there were cases in which some students, from the study of the programming language, were able to infer the underlying concepts of the relevant logic.

Keywords— Computers, episteme, learning, programming, tekne.

I. INTRODUCCIÓN

UNA de las inquietudes recurrentes en el proceso de enseñanza y aprendizaje de la programación de computadores, incluye la discusión acerca de la conveniencia de enfatizar más en los cursos de programación, en la parte puramente conceptual y su fundamento teórico – matemático, o llevar al estudiante directamente al computador y articularlo, desde el principio, con un lenguaje de programación, con el ánimo de que él mismo descubra los elementos teóricos que le subyacen [1]. En otro tiempo, la respuesta pudiera resultar más fácil de encontrar dado que, veinte años atrás, o más, no se contaba con los recursos tecnológicos ni con el acceso a ellos tan simple, sencillo y masivo como sucede hoy.

La aparición de los computadores personales, la accesibilidad a los costos y el cambio de mentalidad que permitió que cada hogar tuviera, por lo menos, un computador (o intentara tenerlo), adicional a la aparición de

¹Producto derivado del proyecto de investigación “Desarrollo de un modelo para la enseñanza y el aprendizaje de la programación en Ingeniería de Sistemas basado en el modelo 4Q y en aprendizaje significativo”, presentado por el Grupo de Investigación en Informática de la Universidad Tecnológica de Pereira.

O. I. Trejos, Universidad Tecnológica de Pereira, Pereira, Colombia, email: omartrejos@utp.edu.co.

Como citar este artículo: Trejos, O. I. Episteme vs. Tekné: una aproximación al aprendizaje de la programación de computadores en Ingeniería de Sistemas, Entre Ciencia e Ingeniería, vol. 13, no. 25, pp. 45-51, enero-junio de 2019. DOI: <http://dx.doi.org/10.31908/19098367.4013>.



ciber salas o café-Internet, ha puesto a la generación actual de estudiantes de la programación en una situación en la cual, encontrar la respuesta invita a procesos investigativos [2] como el que se expone en este artículo.

La inquietud fundamental gira en torno al hecho de pensar que, a partir del aprendizaje y apropiación de los conceptos fundamentales que subyacen a los lenguajes de programación, es decir, la lógica de programación se pueda llegar a tener la proficiencia requerida para entender y utilizar los diferentes lenguajes de programación que existen en el mercado informático de los desarrolladores de software. Idea contrapuesta de la inquietud que surge si, al acompañar al estudiante en un proceso de aprendizaje profundo del manejo de un lenguaje de programación, es posible que él llegue a inferir los conceptos teóricos que le subyacen y, por lo tanto, a extrapolarlos, bien hacia otros lenguajes, o bien hacia otros paradigmas de programación.

El problema no puede resolverse con base en la perspectiva personal del docente de programación, desde su labor pedagógica o desde su experiencia como programador. Al contrario, debe pensarse en diseñar una metodología para obtener conclusiones que orienten la enseñanza y su impacto en el aprendizaje de la programación de computadores, cuando se desarrollan procesos de formación profesional con estudiantes nacidos y criados en medio de un lenguaje natural llamado nuevas tecnologías de la información y la comunicación [3].

Por lo tanto, el propósito de este estudio es el de aportar elementos de juicio que les permitan a los docentes de formación secundaria o involucrados en procesos de formación profesional, enfatizar en los principios conceptuales (*episteme*) de la lógica de programación y en su fundamentación matemática, más que en lo enteramente aplicativo (*tekné*) de un lenguaje de programación [4].

Lo anterior con el fin de que docente elija entre las dos alternativas mencionadas: aprovechar las características de los nativos digitales [5] para profundizar en los procesos de apropiación, asimilación, aplicación, retroalimentación, validación y evaluación del conocimiento; o enfatizar en la formación tecnológica directa y la interacción con los lenguajes de programación para que sea el estudiante quien infiera la teoría que le subyace y pueda aplicarla en otros paradigmas [6].

La novedad del presente artículo subyace en el sentido de que la pregunta por cuál de las opciones es mejor, es recurrente en las discusiones de los docentes de programación, pero pocos estudios se han realizado desde la perspectiva de la investigación educativa, a partir de los cuales, sobre datos veraces y comprobables, se puedan obtener inferencias que conduzcan a conclusiones científicas y menos basadas en la intuición, en la experiencia o en el aparente conocimiento del mismo docente.

De igual manera, el estudio se justifica porque tanto en los colegios y universidades se considera, per se, que la enseñanza de la programación de computadores y sus procesos de aprendizaje asociados, requieren (sin discusión) de la utilización e incorporación de computadores, de lenguajes de programación, de compiladores y de ambientes integrados (IDE por sus siglas

en inglés) que posibiliten, aparentemente, el aprendizaje [7].

¿Se justifica la alta inversión en infraestructura tecnológica para el aprendizaje de la programación, si aún no se ha verificado científicamente que la utilización de los computadores lo promueva y catapulte por un camino más expedito que el estudio detallado y profundo de sus fundamentos teóricos y matemáticos? Esa es la pregunta que se intenta resolver al admitir que, sobre el tema de esta investigación, como se comentará más adelante, puede profundizarse para llegar a conclusiones objetivas dentro del marco de una determinada comunidad de estudiantes y considerando las condiciones particulares que conforman su actual contexto extrainstitucional.

Además del trabajo de campo, en el desarrollo de esta investigación se contó con información obtenida de bibliografía especializada referida, tanto al aspecto teórico investigativo, como a lo práctico y a lo metodológico, así como de la participación del autor en diferentes cursos sobre el tema. A partir del análisis de la información encontrada, se obtuvieron los resultados y se realizaron las inferencias pertinentes. También es importante señalar que esta investigación es producto del proyecto de investigación código 6-16-13, autorizado por la Vicerrectoría de Investigaciones, Extensión e Innovación de la Universidad Tecnológica de Pereira. La investigación se realizó con cursos de 1er semestre de Ingeniería de Sistemas durante los años 2016, 2017 y 2018, en la asignatura Programación I.

En su estructura, el artículo se presenta organizado de acuerdo con el estándar IMRYD [8], que comienza con una Introducción, continúa con la exposición de los elementos teóricos, de la metodología empleada, la explicación de los resultados obtenidos, y finaliza en una discusión que conduce a emitir las conclusiones. En la última parte se relacionan las referencias bibliográficas que sirvieron de soporte para el desarrollo, tanto del artículo como de la investigación.

II. MARCO TEÓRICO

La esencia del conocimiento se ha contrapuesto, desde épocas inmemoriales cuando brillaba la filosofía griega como faro iluminador del pensamiento universal, a la aplicación práctica del mismo [9]. La discusión se centra en la pugna entre el camino para buscar la esencia del conocimiento y llegar a él, por ende, a su apropiación, o aferrarse a la práctica y, a partir de allí, encontrar los elementos teóricos que le subyacen por la vía de la razón heredada de los procesos aplicativos del saber [10]. No es una discusión nueva y por ello, los antiguos griegos llamaron a la esencia del conocimiento *episteme*, y a la aplicación del mismo *tekné*. Estas orientaciones pusieron en escena dos corrientes filosóficas que, con el tiempo, rivalizan en ese anhelo del ser humano de apropiarse y asimilar el conocimiento para mejorarse a sí mismo y a su entorno [11].

La discusión trasciende diferentes épocas del conocimiento, así como diferentes áreas, tal como se reflejó en el siglo VIII cuando los árabes difundieron los

fundamentos del álgebra [12]; en el siglo XVI cuando se plantean las bases de la geometría analítica [13]; en el siglo XVII cuando se propone el cálculo diferencial [14], y en el siglo XIX cuando la matemática toma nuevos rumbos aún no descubiertos [15]. En estos ejemplos, la discusión siempre giró alrededor de la necesidad de definir si es más importante la apropiación y sabiduría alrededor de los fundamentos del conocimiento matemático, o si es aún más relevante saber y conocer en qué situaciones de la vida del ser humano son aplicables, tanto los conocimientos matemáticos ya establecidos, aceptados y universalizados, como los nuevos desarrollos alcanzados a partir de estos.

Hacia la mitad del siglo XX aparecen los computadores como las herramientas que permiten realizar tareas que, para el ser humano, son posibles, pero agotadoras y, por ello, proclives a generar errores en los resultados y en la interpretación [16]. Las computadoras básicamente permiten realizar grandes cálculos con un margen de error establecido por los cánones que dictan las matemáticas, y no por la naturaleza misma de los seres humanos. El hecho llevó a pensar que era posible incluir las computadoras en una inmensa cantidad de tareas que, en sus albores, eran todavía inimaginables.

A partir de la necesidad de encontrar nuevas aplicaciones para los computadores, se desarrolla la lógica de programación como fundamento matemático para capitalizar al máximo las capacidades de procesamiento y cálculo de esta máquina [17]. Al mismo tiempo, se empiezan a desarrollar los lenguajes de programación que son el vehículo para que se instrumentalicen las soluciones que la matemática provee, usando la nueva lógica computacional que acababa de emerger en el conocimiento humano.

Las aplicaciones de los computadores, de la lógica de programación y de los lenguajes de programación fueron cada vez más numerosas y más urgentes para la sociedad. Entonces se inició la especialización del conocimiento en esta materia, al punto de llegar a configurar una nueva profesión que en principio fue un nuevo oficio: el de programador de computadores [18]. Allí apareció, de nuevo, la discusión acerca de lo que necesitaba aprender el programador como base para sus posteriores desarrollos.

De una parte, se identifican quienes defienden el aprendizaje de la lógica de programación representada en sus fundamentos matemáticos, pensando que, si se conoce la esencia del conocimiento, el ser humano puede articular fácilmente sus posibles aplicaciones y desarrollos. De otra parte, están quienes defienden el aprendizaje de los lenguajes de programación, dado que es una nueva era, una nueva sociedad y un nuevo ser humano, cuando se considera posible, y aún se considera, que por el camino del aprendizaje de la *tekné* se puede llegar a los fundamentos que caracterizan su *episteme* [19].

Nunca se pensó, eso sí, que los computadores llegarían a formar parte de la vida cotidiana de los seres humanos, al punto de que aquellas personas que nacen en medio de la tecnología, como sucede en el presente siglo XXI, y para quienes esta se convierte en su lenguaje natural, se conocen como nativos digitales. En cambio, quienes conocieron el mundo sin la presencia de las formas, dispositivos,

productos y servicios tecnológicos, se conocen como inmigrantes digitales, a raíz del impacto que en el día de hoy se experimenta [20] y de la gran incidencia que tienen las modernas redes sociales en la toma de decisiones y en la manipulación de diferentes momentos de la vida.

El cerebro de las personas se encuentra bombardeado por pantallas [21] que en todo momento proveen información desde parámetros objetivos, o como producto de la subjetividad que persigue determinados intereses, tanto en lo social, en lo político, en lo comercial, en lo económico, como incluso, en la esfera puramente personal.

Los anteriores elementos de juicio señalan que la investigación en educación está llamada a buscar respuestas a la inquietud inicial de este documento, considerando las características de los jóvenes de hoy, las bondades que las nuevas tecnologías han incorporado a sus vidas y las limitaciones que ellas mismas proveen dentro de un contexto de interacción social, que se mueve más por canales digitales que por canales directos de interrelación personal [22].

Es lo que persigue esta investigación: intentar encontrar respuestas, o por lo menos tener acceso a elementos de juicio que posibiliten una reflexión objetiva con respecto al aprendizaje de la programación, que contribuya a zanjar la discusión acerca de la relevancia que puede tener la episteme sobre la *tekné* pertinente o, si es del caso, la *tekné* sobre la episteme involucrada. Ambas en el sentido que las aplicaciones, necesidades y requerimientos de la sociedad les exigen a los programadores de computadores que se encuentran en proceso de formación como Ingenieros de Sistemas.

III. METODOLOGÍA

La investigación se realizó durante los años 2016, 2017 y 2018 con los estudiantes de la asignatura Programación I de Ingeniería de Sistemas y Computación de la Facultad de Ingenierías de la Universidad Tecnológica de Pereira. Se escogió el contenido de la asignatura Programación I (que corresponde al paradigma de programación funcional) debido a que sus conceptos innovadores permiten posicionar los grupos en un mismo partidario, a diferencia del curso de asignaturas en donde el contenido era el paradigma imperativo o el paradigma orientado a objetos, pues estas temáticas se reciben como asignaturas durante el proceso de formación en básica secundaria. Lo innovador del paradigma funcional, para los estudiantes de I semestre que recién llegan a la universidad, facilita algunas de las actividades previstas para el desarrollo de la investigación.

Cada grupo se dividió en dos subgrupos, de manera que con uno de ellos se pudiera conducir la asignatura basada en el desarrollo del contenido establecido curricularmente, pero acudiendo exclusivamente a su fundamentación teórica y al modelo matemático que subyace al paradigma funcional, objetivo principal de la asignatura. Con el otro subgrupo se condujo la asignatura interactuando, desde el primer día del curso, con el lenguaje de programación que se ha establecido para tal fin y que corresponde al lenguaje DrRacket. La Tabla I presenta la cantidad de estudiantes

que participaron en la misma y la forma como se estructuraron los subgrupos.

TABLA I
ESTUDIANTES PARTICIPANTES EN LA INVESTIGACIÓN.

Año	Sem	Tot. Est.	SubGrp E*	SubGrp T*
2016	I	23	12	11
	II	20	10	10
2017	I	22	11	11
	II	22	11	11
2018	I	20	10	10
	II	23	12	11
Total		130	66	64
%		100%	50,769%	49,231%

E = Con orientación a la Episteme. T = Con orientación a la Tekné

El contenido, durante el semestre, se dividió temáticamente en 4 módulos que, siendo interdependientes, fueran también independientes en sus linderos conceptuales y que permitieran una evaluación lo más objetiva posible del conocimiento nuevo que se buscaba dejar en cada uno. Cada módulo tomó 4 semanas y así se completaron las 16 semanas que dura el semestre académico. La división conceptual que se realizó fue la siguiente: 1° módulo - Fundamentos, el concepto de función y la estrategia “divide y vencerás”; 2° módulo - Condicionales y Menús; 3° módulo - Recursividad y procesos cíclicos; 4° módulo - Conjuntos de datos (listas y vectores).

El tiempo destinado para cada asignatura se acordó de manera que los estudiantes del subgrupo E pudieran recibir sus clases en horario diferente al subgrupo T, intentando mantener en todo momento las condiciones físicas de ubicación, salón y distribución en ambos grupos, con el ánimo de intentar conservar estas variables lo más aproximadamente posible entre ellas. Las sesiones de cada subgrupo E se realizaron sobre la base de una estrategia de clase magistral en el tablero, enfatizando en la fundamentación matemática asociada al paradigma de programación funcional, que corresponde a la asignatura.

Las sesiones de cada subgrupo T se realizaron algunas veces en sala de computadores, y otras veces con la utilización de computador y videobeam en el aula, pero siempre llevando los conceptos hasta las posibilidades que provee el lenguaje de programación DrRacket. Los talleres desarrollados durante el curso tuvieron una orientación diferente para cada subgrupo: Los talleres que realizaron los estudiantes de los subgrupos E estuvieron orientados a la resolución de problemas desde una perspectiva matemática general y a la luz de la algoritmia conceptual funcional, como herramienta para el planteamiento de las mismas. Los talleres que realizaron los estudiantes de los subgrupos T se orientaron hacia la resolución de problemas utilizando los recursos que el lenguaje de programación DrRacket provee, es decir, bajo el concepto de algoritmia tecnológica que corresponde a la versión de la algoritmia funcional conceptual funcional, pero expresando las soluciones en términos del lenguaje de programación referido.

Se realizaron 4 evaluaciones parciales, todas ellas escritas, y cada una tuvo un peso de 15% sobre la nota final, lo cual suma un 60%. El 40% restante correspondió

al desarrollo de un programa aplicativo, el mismo para ambos subgrupos, que debió sustentarse individualmente y cuyo enunciado exigió una comprensión clara, tanto de los conceptos teóricos como de los recursos sintácticos que provee el lenguaje de programación DrRacket.

La sustentación de dicho programa correspondió a la cristalización del logro de los objetivos planteados en la investigación: determinar si es la episteme un camino para llegar a la tekne, en el aprendizaje de la programación de computadores, o es posible que la tekne posibilite la inferencia de los conceptos que le subyacen, es decir, su propia episteme. Además, la sustentación del trabajo final consistió en realizarle a los estudiantes de ambos subgrupos un conjunto de preguntas previamente escritas (y que ellos desconocían), en las que se cuestionaba tanto lo conceptual como el uso del lenguaje de programación.

Para intentar obtener mayor objetividad en la sustentación, se escribieron previamente 3 preguntas por estudiante, de manera que no se repitieran. Es de anotar que, para ambos grupos, las evaluaciones escritas fueron exactamente iguales en cuanto a su contenido, pero con la libertad de resolverlas por el camino conceptual o tecnológico que consideraran más apropiado, acorde con el proceso compartido con los estudiantes. También vale la pena tener en cuenta que las evaluaciones parciales se realizaron conjuntamente con ambos subgrupos.

Al finalizar el curso se le solicitó a cada estudiante que expresara de manera anónima y con toda libertad, su percepción sobre el proceso llevado a cabo durante el semestre, con el fin de detectar, desde su perspectiva, elementos cualitativos que fortalecieran las inferencias realizadas al respecto. En el desarrollo de este artículo se omitieron intencionalmente ejemplos de talleres y evaluaciones parciales realizadas, así como enunciados finales en virtud de la extensión posible del artículo como tal, y como una forma de respetar el espacio que para este tipo de publicaciones dispone la revista.

IV. RESULTADOS

Para efectos de simplificar los resultados que se obtuvieron durante la investigación, se calculó el promedio de las notas obtenidas por los estudiantes en cada una de las pruebas, tal como se presenta en la Tabla II.

TABLA II
NOTAS OBTENIDAS EN LAS EVALUACIONES PARCIALES.

Año	Sem	Promedio de notas (Eval. Parciales)	
		SubGrp E	SubGrp T
2016	I	4.1	3.7
	II	4.0	3.9
2017	I	4.2	3.8
	II	4.0	3.9
2018	I	4.1	3.6
	II	4.0	3.8
Prom de prom		4.066	3.783
Diferencia		0.283	

La evaluación (trabajo) final que condensaba el resultado de todo el proceso, pues daba la oportunidad de que el estudiante pusiera a prueba los conocimientos aprendidos en programación funcional, arrojó los resultados que se exponen en la Tabla III. En este caso también se acudió al cálculo del promedio de las notas para simplificar la presentación de los resultados.

TABLA III
NOTAS OBTENIDAS EN EL PROGRAMA FINAL.

Año	Sem	Promedio de notas (Trab. Final)	
		SubGrp E	SubGrp T
2016	I	4.6	3.4
	II	4.5	3.5
2017	I	4.3	3.5
	II	4.5	3.6
2018	I	4.7	3.9
	II	4.4	3.6
Prom de prom		4.500	3.583
Diferencia		0.917	

Al finalizar el curso se realizó una prueba anónima, para garantizar libertad y sinceridad en la respuesta, en la cual se les formularon a los estudiantes las siguientes preguntas:

Pregunta 1E. ¿Considera usted que es posible, a partir del estudio de los conceptos básicos que subyacen a la programación funcional y a su modelo matemático, llegar a manejar con total libertad y proficiencia el lenguaje de programación DrRacket? Pregunta 1T. ¿Considera usted que es posible, a partir del estudio del lenguaje de programación DrRacket y de las utilidades que provee, llega a inferir y apropiarse los conceptos que subyacen a su modelo matemático y sus respectivos conceptos básicos? Pregunta 2. ¿Estuvo a gusto en el subgrupo que le correspondió? Pregunta 3. ¿De 1 a 5, en cuánto calificaría su propio nivel de aprendizaje de la programación funcional, de su modelo matemático, de la teoría que le subyace y de sus aplicaciones en el computador?

A los estudiantes que participaron en subgrupos con orientación hacia la *episteme* se les realizó la pregunta 1E, y a quienes formaron parte de los subgrupos con orientación a la *tekné* se les formuló la pregunta 1T. Las otras dos preguntas se formularon a todos los estudiantes, independiente del subgrupo al que pertenecían. Los resultados obtenidos con estas preguntas se presentan en la Tabla IV.

TABLA IV
RESULTADOS OPINIÓN DE ESTUDIANTES.

Año	S	Pregunta 1				Pregunta 2		Pregunta 3	
		1E		1T		Si	No	Sub Grp E	Sub Grp T
		Si	No	Si	No				
2016	I	10	2	8	3	20	3	4.5	3.5
	II	9	1	8	2	16	4	4.0	4.0
2017	I	9	2	7	4	16	6	4.5	3.5
	II	10	1	6	5	18	4	4.0	3.5
2018	I	8	2	7	3	13	7	4.4	3.9

8	II	11	1	8	3	17	6	4.6	3.5
Totales		57	9	44	20	100	30	4.3	3.7
% Relativo		86%	14%	69%	31%	77%	23%	0.683	
% Absoluto		44%	7%	34%	15%				

V. DISCUSIÓN Y ANÁLISIS

En cuanto al tamaño de la muestra, se consideró que seis semestres constituían un período apropiado para realizar los análisis pertinentes en relación con los objetivos de la investigación, teniendo en cuenta que los contenidos curriculares de un programa como Ingeniería de Sistemas, pueden cambiar, y que los jóvenes que vienen de los colegios (pues el estudio se realizó con estudiantes de 1er semestre), a medida que pasa el tiempo traen más capacidades de aprendizaje obtenidas a través de las nuevas tecnologías y, por lo tanto, llegan con más conocimientos nuevos.

Se procuró, en cada semestre, que los grupos se dividieran en subgrupos equivalentes, lo cual se demuestra, como lo presenta la Tabla I, en que del total de estudiantes (130), 66 participaron en subgrupos con orientación hacia la *episteme* y 64 participaron en subgrupos con orientación hacia la *tekné*. Puede observarse en dicha Tabla que la distribución en subgrupos friso el 50%. La separación de los subgrupos se realizó de manera aleatoria con el ánimo de lograr mayor objetividad en los resultados finales.

En cuanto al promedio de las notas parciales, como se presenta en la Tabla II, puede observarse que en todo momento dicho promedio fue superior en los subgrupos con orientación hacia la *episteme* que la orientación hacia la *tekné*, y que la diferencia entre el promedio de promedios fue de 0.283 a favor de los subgrupos orientados a la *episteme*.

Es preciso aclarar que las evaluaciones parciales fueron equivalentes, es decir, se procuró que los enunciados fueran los mismos para ambos subgrupos, con la intención de que cada subgrupo pudiera resolverlo con las herramientas vistas, esto es, los estudiantes del subgrupo orientado a la *episteme* podían resolver los enunciados desde la perspectiva de la algoritmia funcional conceptual, y los estudiantes del subgrupo orientado a la *tekné* podían presentar sus soluciones utilizando el lenguaje de programación DrRacket. Llama la atención de que en todo momento la relación de superioridad en cuanto al promedio de notas de los subgrupos E fue superior al promedio de los subgrupos T.

En cuanto a los resultados que se presentan en la Tabla III, las reflexiones son más significativas puesto que estas notas reflejan el resultado del trabajo final, cuyo objetivo era verificar hasta dónde cada subgrupo de estudiantes era capaz de alcanzar el otro conocimiento, esto es, los estudiantes de los subgrupos orientados a la *episteme* llegan a manejar con proficiencia el lenguaje de programación desde su propia autonomía, y los estudiantes de los subgrupos orientados a la *tekné* llegar a inferir los conceptos que subyacen a la utilización del lenguaje de

programación DrRacket. En esta parte debe tenerse en cuenta que la sustentación oral del trabajo constituyó una parte importante del proceso, y que sus características fueron explicadas en el numeral de Metodología.

De acuerdo con los resultados obtenidos, se observa que el promedio de los promedios de los subgrupos E es notoriamente superior (4.500), en resultados cuantitativos, que el promedio de los promedios de los subgrupos T (3.583), con una diferencia de 0.917 (que podría aproximarse a 1 unidad) a favor del promedio de promedios de los subgrupos E. Este factor parece determinar, para la investigación, que les resultó mucho más fácil a los estudiantes con orientación a la *episteme* adaptarse y hacer suyo el lenguaje de programación DrRacket, que a los estudiantes con orientación a la *tekné* inferir y apropiarse los conceptos teóricos que subyacen al lenguaje funcional de referencia.

Finalmente, en la Tabla IV se obtuvieron unos resultados que se analizan con detenimiento: en primer lugar, la pregunta 1 (en sus dos versiones), tuvo una respuesta mayoritariamente positiva en los estudiantes del subgrupo E que con los estudiantes del subgrupo T. Según los resultados obtenidos, el 86% de los estudiantes de los subgrupos E consideraron que era posible ir de la *episteme* a la *tekné* apropiando y utilizando sus facilidades para construir un programa funcional. Por su parte, y en comparación con estos, solamente un 69% de los estudiantes de los subgrupos T consideraron que el mismo proceso era posible, pero tomando como base la *tekné* para llegar a la *episteme* de la investigación.

En cuanto a la pregunta 2, el 77% de los estudiantes estuvo a gusto con el subgrupo en el cual fue ubicado y solo el 23% respondió NO a dicha ubicación que, como se explicó en un párrafo anterior, fue aleatoria buscando la mayor objetividad posible y sabiendo que los estudiantes que llegan a 1er semestre traen diferentes niveles de conocimiento en programación de computadores.

Los promedios de las notas obtenidas por los mismos estudiantes en cuanto a su nivel de aprendizaje de la programación, que en cierta medida buscaba de manera indirecta responder a la pregunta ¿aprendió a programar?, muestra una valoración mayor en los subgrupos E que en los subgrupos T, al punto que la diferencia entre el promedio de los promedios es de 0.683, un valor significativamente alto considerando que se trata de la tendencia central de un promedio ya calculado. Esto deja presente que los estudiantes de los subgrupos con orientación hacia la *episteme* parecieran haber alcanzado, por lo menos, mayor seguridad en la apropiación de los conceptos, y que eso les confirió un conocimiento que valoraron cuantitativamente, más que aquellos estudiantes que estuvieron en los subgrupos T.

VI. CONCLUSIONES

De los resultados obtenidos en la investigación pueden derivarse varias conclusiones alusivas al propósito planteado. El tamaño de la muestra de estudiantes que participó en la investigación es adecuado, aunque podrían

fortalecerse más los resultados en la medida en que se realice con más estudiantes, lo cual implicaría más cursos por semestre. Debe tenerse en cuenta que, entre los factores exógenos que pueden haber influido en la objetividad de esta investigación, está el hecho de que algunos estudiantes tienen conocimiento previo bien fundamentado en relación con la programación de computadores y, específicamente, con el paradigma funcional; los subgrupos tienen comunicación entre sí durante las demás asignaturas, y los medios de comunicación modernos, así como sus servicios asociados, posibilitan una interacción entre los estudiantes que es prácticamente imposible de evitar.

Se concluye igualmente que una buena estructuración y planeación de un trabajo final, en una investigación de estas características, puede arrojar resultados confiables y objetivos, tal como sucedió en el presente estudio. De la misma forma, se exige una planeación detallada de las evaluaciones parciales para que los resultados sean confiables, en pos de las inferencias posibles que se pueden realizar a partir de las actividades previstas dentro de la investigación.

Además, se cumplió con el objetivo fundamental en un curso de programación y es que, sea por el camino de la orientación hacia la *episteme*, o por el camino de la orientación hacia la *tekné*, los estudiantes que participaron en esta investigación aprendieron a programar bajo el paradigma funcional y aplicando, tanto su modelo matemático (matemática Lambda) como sus expresiones tecnológicas (lenguaje de programación DrRacket).

También se puede concluir que, con relación al aprendizaje de la programación bajo el paradigma funcional, es posible que el estudiante que recibe una orientación epistemológica pueda llegar a apropiarse y aplicar con facilidad los recursos que provee un lenguaje de programación que se ajuste al paradigma estudiado. Igualmente, es posible que, a partir de una orientación enteramente tecnológica de contacto e interacción con un lenguaje de programación desde el primer día de clases, se puedan llegar a inferir los conceptos básicos y el modelo matemático que subyace al paradigma que se esté estudiando, todo ello a la luz del paradigma funcional.

Los resultados cuantitativos obtenidos en la investigación indican que es mucho más expedito el camino para apropiarse el uso del lenguaje de programación a partir de una formación basada en los fundamentos teóricos y conceptuales, que inferir los fundamentos del modelo matemático que subyace a un paradigma de programación a partir del uso y aplicación de los recursos que provee un lenguaje de programación.

REFERENCIAS

- [1] Blanchard, B. System Engineering Management. 4th Ed. New York : Wiley Publish, 2008.
- [2] Bisquerra Alcina, R. Metodología de la investigación educativa. Madrid (España) : Editorial La Muralla S. A., 2004.
- [3] Ballester Valori, A. Meaningful Learning in practice. Islas Canarias : Universitat de les Illes Balears, 2011.
- [4] Campillay S.; Meléndez N. Análisis de impacto de metodología activa y aprendizaje heurístico en asignaturas de ingeniería.. vol. 15

- núm. 22, Actualidades Investigativas en Educación, DOI: 10.15517/AIE.V15I2.18950
- [5] Small, G. Digital Brain. Barcelona : Urano, 2009. p. 254.
 - [6] Van Roy, P. and Haridi, S. Concepts, Techniques, and Models of computer programming. Boston, USA : MIT Press, 2004. p. 936.
 - [7] Ministerio de Educación Nacional. Estándares básicos de competencias en matemáticas. Bogotá : MEN, 2005.
 - [8] Tili, Ahmed, Essalmi, Fathi y et al. Role of personality in computer based learning., Computer in Human Behavior, 64, 2016 págs. 805 - 813.
 - [9] Kline, M.. El pensamiento matemático. De la antigüedad a nuestros días. Madrid : Alianza Editorial, 2012.
 - [10] Acosta Flores, J. Ingeniería de Sistemas: Un enfoque interdisciplinario. Ciudad de México : AlfaOmega Grupo Editor, 2003.
 - [11] Boyer, C. Historia de la Matemática. Madrid : Alianza Editorial, 2010.
 - [12] Strogatz, Steven. The joy of x: from One to Infinity. London : Mariner Books, 2013.
 - [13] Aczel, A. Cuaderno secreto de Descartes. Madrid : Biblioteca Buridan, 2013.
 - [14] Rey Babini, J. y Pastor, J.. Historia de la Matemática Vol. II. Buenos Aires : Gedisa, 2013.
 - [15] Stewart, I. The story of mathematics. London : Quercus Publishing, 2009.
 - [16] Jiménez Murillo, J.. Matemáticas para la computación. Buenos Aires : Alfaomega, 2014.
 - [17] Trejos Buriticá, O. I. Lógica de Programación. Bogotá : Ediciones de la U, 2017.
 - [18] Trejos Buriticá, O. I. Programación Imperativa con Lenguaje C. Bogotá : ECOE Ediciones, 2017.
 - [19] Chukaew, Sasithorn, Panjaburee, Patcharin y et al. A personalized e-learning environment to promote students conceptual learning on basic computer programming. 2014, 5th World Conference on Educational Sciences, págs. 815 - 819.
 - [20] Medina, J. Los 12 principios del cerebro. Bogotá : Grupo Editorial Norma, 2010.
 - [21] Orozco Gomez, G. TvMorfosis. México D. F. : Tintable, 2017.
 - [22] Levy, P. CyberCulture: Electronic Mediations. New York : University of Minnesota Press, 2001.

Omar Iván Trejos Buriticá. Ingeniero de Sistemas. Especialista en Instrumentación Física. Magister en Comunicación Educativa. Phd en Ciencias de la Educación. Universidad Tecnológica de Pereira. Actualmente docente de planta de la Universidad Tecnológica de Pereira, Facultad de Ingenierías, Programa Ingeniería de Sistemas y Computación. ORCID: <http://orcid.org/0000-0002-3751-6014>.