Dificultades en la apropiación del concepto de computabilidad en estudiantes de Ingeniería de Sistemas

Maestría en Educación

Andrea García Rivas

ID: **000708038**

Sublínea de Investigación

Modelos de Acompañamiento para la Formación Integral

Profesor líder

Astrid Viviana Rodríguez PhD

Profesor Tutor

Fabio Andrés Tabla Rico Mg.

Bogotá, D.C. Noviembre de 2020

Dedicatoria.

Un logro que es fruto de muchos esfuerzos, acompañada siempre de mi linda familia: padres, hermana, abuela, tías, primas, primo y mi padrino. A ellos dedicarle el trabajo que espero contribuya en la labor docente del país.

Agradecimientos

Quiero expresar gratitud al director de mi tesis Fabio Andrés Tabla Rico por su dedicación, el apoyo y la paciencia que en todo momento demostró. Al decano de la facultad de ingeniería de la ETITC ingeniero Sócrates Rojas Amador, por el espacio de bits para el desarrollo de la investigación; al docente Eduardo Hernández Ortiz, por sus grandes aportes en el desarrollo del cuestionario del proyecto y finalmente a mis estudiantes del curso Matemáticas Discretas que contribuyeron con su respuesta a los cuestionarios y entrevista los cuales se relacionan a continuación:

Deisy Natalia Silva Narváez, Favio Orlin Alegría viveros, Paula Ximena Vargas Góngora, Andrés Felipe Quintero Sánchez, David Alejandro Navarrete Fernández, Danna Valentina Coy García, Javier Felipe Sánchez Infante, Cesar Oswaldo Rodríguez Suarez, Esteban Sánchez González, Exalus Ruth Bibiane, Randy Joel Caballero López, Erick Santiago García Sánchez, Nicol Valentina Alvis Buitrago, Giovanny Alexander Caicedo Cruz, Harold Giovanny Flórez Rubiano, Sergio Nicolás Gutiérrez Suarez, Erick Ferney Delgado Montañez, Luis Miguel Mesa García, Juliana Valentina Cubillos León, Laura Andrea Gómez Agudelo, Harrison Daniel Ramírez Burgos, Mauricio Andrés Martínez Rada, Camilo Alberto Torres Lozano, Steven Alexander Rodríguez Díaz y Lucner Joseph.

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS -UNIMINUTO- MAESTRÍA EN EDUCACIÓN	
RESUMEN ANALÍTICO ESPECIALIZADO -RAE-	
1. Información General	
Tipo de documento	Tesis de grado.
Programa académico	Maestría en Educación, metodología distancia, modalidad virtual.
Acceso al documento	Corporación Universitaria Minuto de Dios – UNIMINUTO.
Título del documento	Dificultades en la apropiación del concepto de computabilidad en estudiantes de ingeniería de sistemas.
Autor(es)	Andrea García Rivas
Director de tesis	Astrid Viviana Rodríguez Sierra
Asesor de tesis	Fabio Andrés Tabla Rico
Palabras Claves	Computabilidad, funciones recursivas, matemática discreta, lógica computable, máquina de Turing, algoritmia, didáctica.

2. Descripción

La investigación con enfoque mixto presenta la relación entre los niveles de apropiación del concepto de "computabilidad" y las dificultades que experimentan los estudiantes del curso de "matemáticas discretas". Desde el análisis cuantitativo se determinaron mediante una prueba de conocimiento los niveles: alto, medio y bajo, desarrollando así el análisis cualitativo el cual mediante una entrevista a un grupo focal se describen las dificultadas asociadas a la apropiación del concepto de "computabilidad"; con ello proponiendo una serie de lineamientos que orienten la enseñanza del mismo. El estudio se desarrolló en la ciudad de Bogotá en la institución de educación superior Escuela Tecnológica Instituto Técnico Central (ETITC) en el primer semestre del 2020, con estudiantes pertenecientes al programa de ingeniería de sistemas.

3. Fuentes

Brousseau, G. (1986). Fondements et méthodes de la didactiques des mathématiques. *Recherches en didactique des Mathématiques.* 7(2), 33-115.

Castro, I. (2011). *Máquinas de Post. Apuntes de Clase*. Editorial Universidad Nacional de Colombia.

Copeland, B. J., Posy, C. J., & Shagrir, O. (2013). *Computability: Turing, Gödel, Church, and beyond*. The MIT Press.

Freund, M. (2011). Lógica, matemáticas y conceptualismo. *Signos filosóficos*, 13(25). Recuperado de http://www.scielo.org.mx/scielo.php?scriptsci_arttext&pid=S166513242011000100001&lng-=es&tlng=es.

Guzmán, M. (2007). Enseñanza de las Ciencias y la Matemática. *Revista Iberoamericana de Educación*, (43). 19 – 58. Recuperado de https://rieoei.org/historico/documentos/rie43a02.pdf

Hawking, S. (2006). *Dios creo los números. Los descubrimientos matemáticos que cambiaron la historia*. 993-1024. Bogotá: Planeta.

Leavitt, D. (2006). *El hombre que sabía demasiado. Alan Turing y la invención de la computadora.* Barcelona: Antoniobosch.

Mota, S. (2018). Dos concepciones del lenguaje: Wittgenstein y Chomsky en torno a la recursión como "buena" explicación de la naturaleza humana. *Praxis Filosófica*, (46), 125-149.

Penrose, R. (1989). *La mente nueva del emperador. En torno a la cibernética, la mente y las leyes de la física*. México DF: Fondo de Cultura Económica.

Trejos, O. (2015). Algoritmo recursivo diferente para hallar elementos de la serie de Fibonacci usando programación funcional. *Avances Investigación en Ingeniería.* 11(2), 19-28.

Trejos, O. (2017). Metodología algorítmica para construir funciones que resuelvan cálculos basados en procesos simples usando programación funcional. *Investigación en ingeniería.* 63-76. DOI: doi.org/10.1804/avances. v14i1

Turing, A. (1936). On Computable Numbers, with application to the Entscheidungsproblem. 2(42), *Proceedings of the London Mathematical Society*, 2 (42), 230-265. Recuperado de https://doi.org/10.1112/plms/s2-42.1.230

Turing, A. (1937). On Computable Numbers, with an Application to the Entscheidungsproblem: A correction. *Proceedings of the London Mathematical Society*. 2(43), 544-546.

Vilca, C. (2016). *Máquinas de Turing y sus aplicaciones.* ISSN 2415-2323

Xu, Zhi-Wei & Tu, Dan-Dan. (2011). Three new concepts of future computer science. *Journal of Computer Science and Technology*. Beijing: Springer Nature. 26(6). 616-624. Recuperado de http://dx.doi.org/10.1007/s11390-011-1161-4

4. Contenidos

El contenido del documento se compone de cinco capítulos:

Capítulo 1. Planteamiento del problema
Capítulo 2. Marco referencial
Capítulo 3. Metodología
Capítulo 4. Resultados y análisis
Capítulo 5. Conclusiones

5. Método de investigación

La investigación "dificultades en la apropiación del concepto de computabilidad en estudiantes de ingeniería de sistemas" se desarrolló bajo el enfoque mixto secuencial, puesto que se buscó un acercamiento a identificar los niveles de apropiación en computabilidad con una prueba de conocimiento y conocer por parte de los estudiantes las dificultadas asociadas en la comprensión del concepto de computabilidad, mediante una entrevista a un grupo focal. Dentro del análisis mixto el alcance fue correlacional debido a que se relacionaron los resultados del análisis cuantitativo con el cualitativo mediante unas matrices que describen de manera global las dificultades en cada variable propuesta. Finalizando en una serie de lineamientos en la enseñanza de la "computabilidad".

6. Principales resultados de la investigación

Los hallazgos principales en la investigación se presentan a continuación:

Se determinaron los niveles de apropiación del concepto de "computabilidad" en los estudiantes del curso "matemáticas discretas" del programa de Ingeniería de Sistemas. Los cuales fueron: alto (19%), medio (62%) y bajo (19%). Presentando los niveles de manera global y de cada categoría en las que se dividió la investigación.

Se identificaron las dificultades asociadas a la apropiación del concepto de “computabilidad”, a partir del reconocimiento de la voz de los estudiantes del curso “Matemáticas discretas” del programa de Ingeniería de Sistemas, mediante el uso del software Atlas.Ti.

Se plantearon una serie de lineamientos que orientaran la enseñanza de la “computabilidad”, a estudiantes del programa de Ingeniería de Sistemas, con cambios en el currículo e introducción de temas propios del que hacer de la ingeniería de sistemas.

7. Conclusiones y Recomendaciones

Las siguientes son las principales conclusiones y recomendaciones:

Las principales dificultades de la apropiación del concepto de “computabilidad” provienen de “la matemática de la computabilidad” y específicamente tiene que ver con la comprensión de los conceptos matemáticos, el simbolismo y la algebrización.

En los niveles de apropiación en general se obtuvo un nivel medio del orden del 62%, donde se hace necesario profundizar en las variables “máquina de Turing” y “funciones recursivas”.

Es posible la enseñanza de la computabilidad desde los primeros semestres con nuevos enfoques y no solamente desde la perspectiva tradicional y así tener una perspectiva de lo que hoy significa la “computabilidad”. Rompiendo el compartimiento desde cada una de las temáticas del curso y profundizando en el isomorfismo entre la lógica proposicional, teoría de conjuntos, circuitos lógicos y álgebra Booleana.

Elaborado por:	Andrea García Rivas
Revisado por:	Fabio Andrés Tabla Rico
Fecha de examen de grado:	

Índice

Índice de Tablas

Índice de figuras

Índice de Apéndices

Resumen

El proyecto de investigación “dificultades en la apropiación del concepto de computabilidad en estudiantes de ingeniería de sistemas” se realizó en la ciudad de Bogotá, durante el primer semestre del año 2020, en la institución de educación superior Escuela Tecnológica Instituto Técnico Central “ETITC”, en la asignatura “matemáticas discretas”.

El universo muestral estuvo conformado por 26 estudiantes. El enfoque mixto de la investigación tiene un diseño secuencial y su objetivo fue analizar la relación entre los niveles de apropiación del concepto de “computabilidad” y las dificultades que experimentan los estudiantes alrededor de cuatro variables.

Una de las principales dificultades en la apropiación del concepto de computabilidad proviene de “la matemática de la computabilidad” y específicamente está relacionado con la comprensión de los conceptos matemáticos, el simbolismo y la algebrización; también se requiere la visión holística del isomorfismo existente entre la teoría de conjuntos, la lógica proposicional, los circuitos lógicos y el álgebra Booleana. Los resultados muestran que las dificultades caracterizadas son próximas a los niveles de apropiación (identificados) alrededor del concepto de “computabilidad”.

Las limitantes se correspondieron con la crisis generada por el Covid19, donde no fue posible realizar la aplicación de los instrumentos de manera presencial.

La enseñanza de la matemática en nuestros tiempos no será la misma de antes, donde el énfasis estaba en la matemática continua; hay una nueva mirada hacia lo discreto y lo finito.

Finalmente, se plantean una serie de lineamientos para la enseñanza matemática del siglo XXI en las facultades de ingeniería.

Palabras claves: Computabilidad, matemática discreta, funciones recursivas, máquina de Turing, lógica computable, algoritmia.

Abstract

The research project “difficulties in the appropriation of the concept of computability in systems engineering students” realized in the city of Bogotá, during the first semester of 2020, at the higher education institution Escuela Tecnológica Instituto Técnico Central “ETITC”, in the subject "discrete mathematics".

The sample universe consisted of 26 students. The mixed approach to the research has a sequential design and its objective was to analyze the relationship between the levels of appropriation of the concept of “computability” and the difficulties experienced by students around four variables.

One of the main difficulties in the appropriation of the concept of computability comes from "the mathematics of computability" is specifically related to the understanding of mathematical concepts, symbolism and algebrization; the holistic view of the isomorphism existing between set theory, propositional logic, logic circuits and Boolean algebra is also required. The results show that the difficulties characterized are close to the levels of appropriation (identified) around the concept of “computability”.

The limitations correspond to the crisis generated by Covid19, where it was not possible to apply the instruments in person.

The teaching of mathematics in our times will not be the same as before, where the emphasis was on continuous mathematics; there is a new look at the discreet and the finite.

Finally, a series of guidelines are proposed for the mathematics teaching of the 21st century in engineering schools.

Keywords: Computability, discrete mathematics, recursive functions, Turing machine, computable logic, algorithmic.

Introducción

Las computadoras en el siglo XXI son la representación de una revolución histórica y social, las matemáticas la ciencia detrás de estas máquinas, la “matemática discreta” su rama especializada, y la computabilidad la piedra angular de esta dinámica.

La computabilidad es un concepto globalizante, puede equipararse al dominio del fuego o el descubrimiento de la rueda; una vez se tiene el control del fuego este sirve para hacer alimentos, cuidarse de las fieras o alumbrar las cuevas; igual sucedió con los computadores, inicialmente procesar números, luego hacer documentos, enviar cartas (mail), hacer una reserva de tiquetes, saber cuánto dinero se tiene en el banco o conocer el estado de salud de una persona.

En este trabajo se hace el estudio de la “computabilidad”, pero no solo desde el punto de vista matemático, de su formalismo y aplicabilidad; existe otra arista en este proyecto de investigación, una interpretación novedosa de la didáctica en cuanto a las dificultades de la apropiación, conceptualización y operabilidad del concepto de computabilidad.

La década del 30 del siglo pasado fue la edad de oro en el logro de la formalización del concepto de computabilidad, dicha trazabilidad se puede encontrar en los *paper* de los más importantes centros de investigación universitaria en Europa y Estados Unidos; más sorprendente aún, es lo jóvenes que eran estos matemáticos, la mayoría veinteañeros y la forma casi independiente y simultánea que hicieron las cinco propuestas de computabilidad más relevantes: máquinas de Turing, cálculo lambda, funciones recursivas, máquinas de Post y algoritmos de Markov; todas formalmente equivalentes.

Hay una pregunta que se puede hacer solo retrospectivamente: ¿cuál es la razón para que el modelo de Turing sea el más conocido y relevante?; ¿cuál es la diferenciación con los otros modelos? La respuesta que se intenta dar: *“La explicación didáctica de dicho modelo”*, y es en este sentido que se hace la consideración de la principal razón de su éxito y uno de los argumentos fundamentales que se trata de demostrar en este trabajo de investigación.

Pero no es solo hacer una mirada a lo que sucedió hace 80 años en los maravillosos años 30 del siglo XX; es también poner énfasis a la didáctica en el presente. En consecuencia, este trabajo de grado realizó además del anterior otro esfuerzo fundamental: diagnosticar las dificultades en la apropiación del concepto de computabilidad, el cual es transversal al currículo de ingeniería de sistemas y de acuerdo con la especificidad de la enseñanza de la "matemática discreta" en ingeniería de sistemas; ingeniería que busca dar soluciones de computabilidad a los problemas inherentes del tratamiento de la información. Todo lo anterior, con el fin de diseñar estrategias, procesos de formación continua y avanzada apoyando el desarrollo de competencias que requieren los ingenieros de sistemas y así permitirá afrontar de una mejor manera los retos universales del desafío científico - técnico en Colombia.

En el capítulo primero se presenta el problema de investigación; se hace una aproximación a la "computabilidad" como el eje de trabajo, y dentro de este contexto se establece como objetivo general, analizar la relación entre los niveles de apropiación del concepto de "computabilidad" y las dificultades que experimentan los estudiantes en el proceso cognitivo, lo que permitirá plantear los objetivos específicos para determinar los niveles de apropiación, describir las dificultades y plantear una serie de lineamientos que orienten de una forma más adecuada la enseñanza de la computabilidad en estudiantes de ingeniería de sistemas.

El capítulo segundo, es el marco referencial; donde se presenta el contexto teórico de la computabilidad y aquí la didáctica vuelve y juega su papel, a pesar de la complejidad y alto nivel matemático que requiere el tratamiento de estos temas se debe hacer un esfuerzo para su mejor entendimiento. La computabilidad requiere de la matemática en general y de la "matemática discreta” en particular, para dimensionar la apropiación de este concepto en cada una de sus principales áreas temáticas como son: lógica, teoría de conjuntos, teoría de números, álgebra booleana, grafos y árboles.

El tercer capítulo desarrolla la metodología de la investigación, se explica el enfoque mixto, el alcance explicativo del estudio y su diseño secuencial; se describe la población, la muestra, los instrumentos de recolección de datos y su método de análisis.

En el cuarto capítulo están los resultados al analizar la relación entre los niveles de apropiación del concepto de computabilidad y las dificultades que experimentan los estudiantes.

Las conclusiones son el último capítulo, el capítulo cinco; donde están los principales hallazgos, las limitaciones y las nuevas ideas para futuras investigaciones que surgen como producto de este trabajo.

En un mundo tan cambiante es difícil predecir el futuro; hoy se puede afirmar y reconocer el papel que ha mantenido las matemáticas tanto en el pasado como en el presente para lograr los grandes avances en los desarrollos de la ciencia y la tecnología. Por ello se hace necesario considerar los contenidos porque la enseñanza matemática no será la misma de antes; hay una nueva mirada hacia lo discreto y lo finito, desplazando la matemática del continuo y este proceso será aún mayor en profesiones relacionadas con tecnologías de la información y las comunicaciones.

La presente investigación es el producto del trabajo de grado de la -Maestría en Educación- de la Corporación Universitaria Minuto de Dios durante los años 2019 – 2020; el trabajo de campo se realizó en la Escuela Tecnológica Instituto Técnico Central -ETITC- de la ciudad de Bogotá, institución pública de formación universitaria en el programa curricular de Ingeniería de Sistemas, en la asignatura de Matemáticas Discretas.

Capítulo 1. Planteamiento del problema

1.1 Antecedentes

En el siglo XXI se vive lo llamado por algunos la “era de la información”, la omnipresencia de las computadoras, los algoritmos y las aplicaciones en la vida cotidiana; algunos autores han dado en llamar la “cuarta revolución tecnológica”, y dentro de este proceso existe un concepto que pareciera “invisible”, pero se encuentra inmerso en toda su dinámica, el de “computabilidad”.

El concepto mismo de computabilidad es intuitivo y problemático. A pesar de los esfuerzos hechos por matemáticos, ingenieros, científicos y filósofos, no se ha terminado de construir y lo que es más grave aún, su aplicabilidad no se puede restringir a un área específica de la ingeniería de sistemas, la informática o las ciencias de la computación; es una exigencia social en los diferentes contextos de aplicación práctica del conocimiento humano.

La tecnología ha sido inherente a la cultura humana; se pueden rastrear los avances de la computabilidad desde el mismo hueso de Ishango con 20.000 años de antigüedad, en las tablas de arcilla de la cultura sumeria, en las pirámides egipcias o también en la clásica cultura griega con su enorme legado.

Pitágoras estableció la frase: “el número es la medida de todas las cosas”; los árabes el álgebra en el siglo IX de nuestra era, el cuadrivio de la edad media, Leibniz buscó expresamente un procedimiento general para resolver las demostraciones y el portentoso desarrollo analítico de la matemática en los últimos 300 años.

El siglo decimonónico de las emergentes ciencias naturales marca indeleblemente los rumbos de la computabilidad con el proyecto de la máquina analítica de Charles Babagge en 1837, casualmente esta fecha antecede en 100 años a los famosos paper de Turing y Church; en 1847 se publica el libro: "Análisis matemático de la lógica", cuyo autor George Boole es un autodidacta haciendo de la lógica una lectura algebraica y finitista.

En la transición del siglo XIX al siglo XX hay un optimismo que se refleja en el programa formalista de Hilbert cuando plantea sus famosos 23 problemas en la conferencia de París del Congreso Internacional de Matemáticos del 8 de agosto del año 1900 en la Universidad de la Sorbona. En particular, el décimo problema: "Encontrar un algoritmo que determine si una ecuación diofántica polinómica dada con coeficientes enteros tiene solución entera", buscar la solución a este problema, fue la veta que hizo posible concretizar la formalización del concepto de computabilidad en los años 30.

Otra manera formal de plantear el famoso décimo problema de Hilbert, que es uno de los 23 que propuso, y plantearlo en términos de ecuaciones diofánticas es lo siguiente: *Dar un algoritmo que permita decidir para un polinomio dado* $\mathbf{q}(X_1,X_2,\ldots,X_n) \in \mathbb{Q}\,[X_1,X_2,\ldots,X_n]$ *si existe un punto* $(x_1, x_2, \ldots, x_n) \in \mathbb{Z}^n$ tal que: $\mathbf{q}(x_1, x_2, \ldots, x_n) = \mathbf{0}$

Es importante anotar que la visión optimista de comienzos del siglo XX no es exclusiva de Hilbert con su programa formalista, el logicismo de Bertrand Russell también lo era; en otras áreas del conocimiento como la química con el descubrimiento del electrón por Thompson en 1897; en la física con la máquina de vapor, o en electricidad con las ecuaciones del Maxwell.

Desde el punto de vista de la computabilidad, es muy importante advertir sobre las diferencias de la forma de cómo se construyó el conocimiento en las llamadas ciencias naturales como la física o la química; este saber se hace desde la interpretación de la realidad; se observa el fenómeno que sucede en la naturaleza, y a partir de ahí se construye un modelo. La computabilidad se construyó de forma totalmente contraria a este modelo; primero fue su conceptualización y luego la concretización de las máquinas que computan.

En el VIII Congreso Internacional de Matemáticas - Bolonia (Italia) del año 1928, Hilbert vuelve 28 años después sobre la misma pregunta del décimo problema de 1900, pero abordándolo de manera diferente incluso le pone un nuevo nombre con el que permanecerá en la historia.

> Uno de tales problemas era el de la decisión (Entscheidungsproblem), la cuestión de si existe un método eficaz para decidir, dada cualquier fórmula bien formada del cálculo de predicados de primer orden, si es o no válida en todas las interpretaciones posibles. Eso se había resuelto en afirmativo para determinadas clases especiales de fórmulas, pero el problema general seguía abierto en esa época. (Alonso, 2013, p.94).

Posterior a los años 30 se emprende el titánico esfuerzo de concretizar el computador, la idea ya había quedado plasmada, el concepto ya estaba formalmente elaborado; y en la década del 40 dieron a luz las primeras computadoras analógicas tanto en Europa como en Estados Unidos, esta máquina denominada “computador” fue el resultado de una idea, de un pensar, de una abstracción.

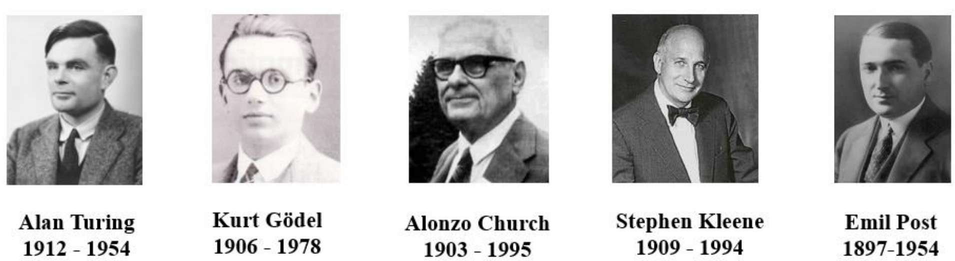
La teoría de la información de Claude Shanon (1948) y la invención del transistor en ese mismo año, consolidaron este proyecto. Hacia la década el 50 se construyen las primeras computadoras digitales que significó la reducción en costo y tamaño. El refinamiento de la programación lógica en los años 60 y la continua miniaturización de los dispositivos electrónicos en la década del 70 hizo posible para la empresa IBM tener un computador de tamaño personal para los 80, logrando la popularización de este artefacto. La década del 90 trajo internet y su masificación, el siglo XXI las redes de 5ª generación.

Es importante resaltar como en cuanto a la formalización del concepto de lo computable, su tiempo histórico se enmarca en la década del 30 del siglo XX, cuando se propusieron nuevas caracterizaciones de la idea de “computabilidad algorítmica”, cuyas propuestas fueron realizadas por un grupo de jóvenes matemáticos y lógicos.

Este relato, es un ejemplo para los jóvenes y la historia: además de sorprendente la edad que tenían cuando se hicieron estos grandes aportes, como prueba de ello se pone entre paréntesis la edad cronológica del momento que escribieron sus más importantes paper: Alan Turing (24 años), Kurt Gödel (25 años), Alonzo Church (33 años), Stephen Kleene (26 años) y Emil Post (33 años) entre otros y el padre (académico) de todos ellos David Hilbert ya contaba con 74 años, estaba jubilado y el nazismo había destruido su paraíso, la escuela matemática que había construido durante toda su vida en Götingen (Alemania).

Tal como se muestra en la figura 1, ellos fueron los protagonistas y jóvenes, no solo cambiaron la matemática; sus ideas fueron transformadoras, cambiaron el mundo; en contraposición a las más encumbradas teorías y en algunos casos a los más reconocidos científicos del momento.

Figura 1. Los protagonistas de la generación del 30 en el siglo XX



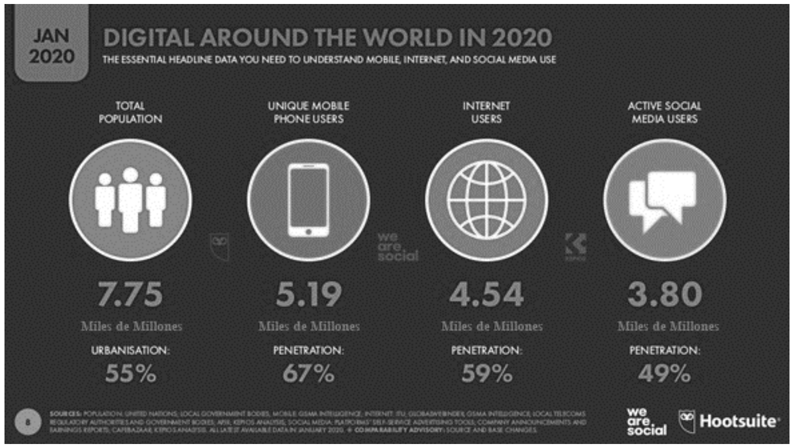
Fuente: construcción de la autora

1.1.1 La magnitud de las cifras

La magnitud de las cifras es enorme, la población mundial para el año 2020 es de 7.750 millones de personas; datos conservadores ubican en 50% la población mundial que utiliza máquinas de

Turing (llámense computadores, celulares o tabletas); estos datos son tomados según el informe que presenta "We Are Social y Hootsuite".

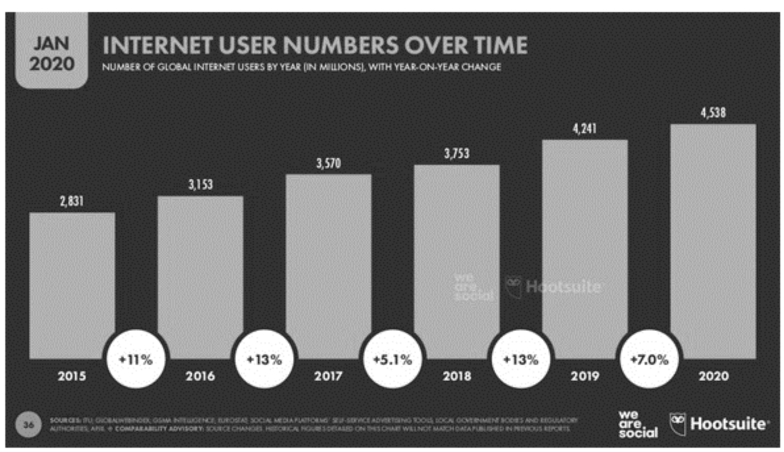
Figura 2. Cifras globales de lo Digital en el Mundo año 2020



Fuente: tomado de la página "wearesocial"

Otro referente para tener una aproximación al número de máquinas de Turing que hoy existen en el mundo es el número de usuarios de internet. La cifra para el año 2020 es de 4.540 millones de personas, 59% de la población mundial, son las que se conectan con internet; ello permite deducir que existen por lo menos igual número de máquinas de Turing.

Figura 3. Crecimiento de los Usuarios de Internet en el período 2014 - 2020



Fuente: tomado de la página "wearesocial"

El curso de “Matemática discreta” hace parte de la malla curricular del programa Ingeniería de Sistemas en la ETITC; esta asignatura es uno de los primeros encuentros de los estudiantes con el concepto de computabilidad, pone en evidencia las falencias en los conceptos matemáticos básicos, ya que se presentan dificultades en la comprensión de los conocimientos necesarios de sus temáticas las cuales requieren un alto nivel de formación y formalización en matemática; lo mencionado anteriormente se ha evidenciado mediante test, talleres, exámenes, ejercicios realizados en el aula, los cuales se han realizado en los últimos cinco años, durante 10 semestres académicos y se encuentran anexos al final del documento. Por ello Bourque y Fairley (2014) propone “establecer en el diseño curricular de programas relacionados con ingeniería de software; ubicar «matemáticas discretas» en el segundo año universitario, porque considera necesario tomarlo a posteriori de los cursos de cálculo y álgebra” (p.86).

Quevedo, (2009) expresa: “En la actualidad, los docentes universitarios del área de ingeniería se preocupan por lo que deben enseñar en el aula, pero no por la manera como se transmite el conocimiento” (p.3). La enseñanza tradicional como lo afirma Rodríguez (2013), “se basa en la estandarización del conocimiento y métodos tanto directivos como autoritarios que conciben al maestro como el ejecutor de normas y al estudiante como un sujeto pasivo” (p.5); es por ello que surge un problema en el contexto de la educación superior, en la enseñanza de la matemática en ingenierías.

De acuerdo con Solar, García, Rojas y Coronado (2014), “en los cursos de formación universitaria se ha evidenciado que la corriente didáctica tradicional es la protagonista, conllevando un currículo poco flexible en términos de innovación para la enseñanza de las matemáticas”. (p.43)

La enseñanza de la matemática es difícil en sí misma y en particular de las "matemáticas discretas" como se expresa en el siguiente texto:

> El aprender a construir procedimientos que permitan calcular, es de gran importancia en el proceso de la enseñanza del aprendizaje, porque se pasa del adiestramiento, al entendimiento de los procesos lógico - deductivos que hay tras dichos cálculos. Comprender el significado de algoritmo, computabilidad, recursividad y computabilidad algorítmica entre otros. (Castro, 2011, p.89)

1.2 Formulación del problema de investigación

Los avances en ciencia y tecnología exigen nuevas necesidades para poder incorporar este potencial en el desarrollo de la sociedad; la masificación de los computadores a nivel mundial a partir de los años 80 del siglo pasado planteó nuevos problemas de carácter científico-técnico.

Uno de los temas de estudio por los grandes centros de investigación mundial es la computabilidad; los últimos conocimientos en la tecnología de esta materia están siendo reservados por dichos centros de investigación; donde a diferencia de hace 80 años ya no son las universidades como la Universidad de Princeton (EEUU) y la Universidad de Cambridge (Inglaterra) las que tenían el protagonismo; en la actualidad son centros privados de las grandes corporaciones tecnológicas con enfoques de investigación y desarrollo (I + D), logrando patentes que generan enormes ganancias en sus resultados porque es mucho el dinero que está en juego en este tipo de decisiones.

Hay una diferencia entre la investigación de frontera en computabilidad y el conocimiento base de estos desarrollos el cual es matemático; la matemática asociada a la tecnología es conocimiento de todos y afortunadamente por ahora todavía tiene carácter público. Dentro de este contexto se entenderá porque los currículos asociados a esta especialidad; los contenidos de su enseñanza son de libre acceso y no tienen la condición de ser secretos y de alguna manera todos los países han asumido la responsabilidad de impartir un conocimiento y enseñanza en este tipo de conocimiento; el cual es muy homogéneo.

La computabilidad tal como se entiende hoy, es un resultado de la modernidad, pero no es nueva; el concepto de algoritmo en el siglo XXI lleva la impronta de Euclides (-325 a. C. -265 a. C.) con su algoritmo para calcular el máximo común divisor de dos números; es decir, se tiene más dos milenios y ha sido transversal en todo este tiempo. La cultura de la Mesopotamia, la cultura egipcia, la cultura china, la cultura árabe en el siglo IX con el álgebra, la edad media, el renacimiento y los siglos XVIII y XIX ayudaron en su consolidación.

Actualmente, la "Máquina de Turing" que es una abstracción matemática, tiene el mayor reconocimiento de la formalización del concepto de computabilidad, esta definición es evidente por sí misma en el mundo de lo real a pesar de estar definida desde el mundo de lo formal, de la matemática; por ello es necesario su estudio y análisis a partir del paper elaborado por A. M. Turing en 1936 el cual lleva por título: "Sobre números computables con una aplicación al problema de la decisión".

Si bien el concepto de recursividad se trató de asociar al concepto de computabilidad en aquellos años 30; para fines del siglo XX, ya dichos conceptos tenían vida propia y no tienen igual significado, como lo menciona

> El concepto de recursión se ha aplicado con cierta ambigüedad dentro de la teoría de la computabilidad. Dicha noción se comenzó a aplicar para significar 'computabilidad'. Sin embargo, el término 'computabilidad' (o 'efectivamente computable') se refiere a una función para la que existe un procedimiento (efectivo) mecánico finito (esto es, un procedimiento generativo/computacional o algoritmo) por medio del cual es computada o calculada; esto es, si se ha definido para ella una secuencia de reglas con las que, dado un input, se obtuviera un valor o output, a través de una secuencia finita de operaciones. (Mota 2018, p.159).

La computabilidad definida por Turing se complementa con la teoría de funciones utilizada en el "Teorema de incompletitud" de Kurt Gödel, con el cálculo lambda de Alonzo Church y el trabajo de Stephen Kleene en cuanto a funciones recursivas.

Las dificultades en la enseñanza de las matemáticas en general y de la computabilidad en particular, plantea una búsqueda y ese será el objeto de este trabajo de maestría.

Se propone identificar las dificultades asociadas a la apropiación del concepto de "computabilidad" en estudiantes de ingeniería de sistemas; este es entre otros, uno de los retos que se proponen en la formación ingenieril en el siglo XXI.

En resumen, surge la necesidad de plantear nuevos contenidos y métodos de enseñanza, donde se logre sin perder el rigor matemático que el estudiante de ingeniería de sistemas del ETITC comprenda de manera global y puntual el concepto de computabilidad; lo cual deja la siguiente pregunta:

Pregunta General

¿Cómo se relacionan los niveles de apropiación del concepto de "computabilidad" con las dificultades que experimentan los estudiantes del curso de «matemáticas discretas» del programa de Ingeniería de Sistemas en la Escuela Tecnológica Instituto Técnico Central – ETITC?

Preguntas específicas

- ¿Cuáles son los niveles de comprensión en la apropiación del concepto de computabilidad en estudiantes de Ingeniería de Sistemas?
- ¿Cuáles son las dificultades asociadas a la apropiación del concepto de computabilidad en los estudiantes de Ingeniería de Sistemas?
- ¿Qué debería considerarse en la enseñanza de la "computabilidad" a estudiantes de ingeniería de sistemas?

1.3 Justificación

La computabilidad, si bien surge desde las matemáticas mismas, también ha influenciado a la matemática; la relación hoy es bidireccional con autonomía en cada uno de los dos campos. Así mismo, la computabilidad ha influenciado a la sociedad en general; está presente en la industria, en internet, en las comunicaciones, en la telefonía móvil, salud, educación, entre otros. Pero esa influencia en la gran mayoría de la población es un modelo de consumo de usuario final, donde se da más importancia a la funcionalidad que a tratar de entender el conocimiento asociado a dichos procesos.

Deconstuir la "caja negra" de la tecnología es una responsabilidad de todos: del gobierno, las universidades, la academia, los profesores y los estudiantes, de la sociedad en general; en el mundo global ser simples consumidores de tecnología no lleva al desarrollo de país, no crea una comunidad científica – técnica que esté al tanto de la ciencia a nivel mundial.

En la actualidad se viven dos características las cuales parecieran ser paradójicas pero que a su vez hacen parte de la misma moneda; por un lado, es una "era computacional" donde estas máquinas inundan cada vez más de forma avallasadora la vida y la cotidianidad; y por la otra se da por sentado muchos conceptos asociados a esta era tecnológica, donde se piensa: "que solo es hundir un botón" (mitos urbanos) con la característica de nuestros tiempos, la computación.

Por lo tanto, la elaboración de este proyecto responde a la necesidad de abordar el tema de la computabilidad desde una perspectiva universal, global e histórica, que aporte a los estudiantes la importancia de conocer, entender y aplicar la computabilidad en las áreas de su formación como son: desarrollo de software, lenguajes de programación, construcción de algoritmos y por otra ofrecer una alternativa de cambio para el mejoramiento de la educación en los estudiantes de ingeniería de sistemas en el curso de “matemáticas discretas”, curso que se ha dictado en los últimos cinco años, durante diez semestres académicos.

Los estudiantes en general reciben la enseñanza de las matemáticas de manera general sin un contexto de especificidad para su formación disciplinar, hacer explícito la relación histórica y actual entre las matemáticas y la computabilidad será de gran utilidad para qué los alumnos entiendan la importancia de la necesidad de la formalización en la comprensión de los temas correspondientes a lo que va a ser su desempeño profesional.

Esta es una responsabilidad desde la institucionalidad, desde lo que debe ser una política educativa, un esfuerzo principalmente de los directivos de universidades, del profesorado que asume la tarea directa con los educandos y de los estudiantes que están en su proceso de formación. Tampoco se puede desconocer que algunas veces los estudiantes toman las materias de matemáticas como una materia más para cumplir con un requisito de la malla curricular.

La contextualización específica no es solo un problema de los currículos, ni de lo que sucede en el diario acontecer del aula de clase, en la mayoría de libros guías de “matemáticas discretas” tanto nacionales como internacionales este tema se aborda como otro capítulo más de las matemáticas cuando lo discreto toma relevancia en el mundo de lo computacional; incluso en teoría de números se sigue trabajando sobre esquemas de la matemática continua donde la matemática discreta parecieran no existir en el nuevo horizonte de la matemática computacional. Con ello, es importante establecer las dificultades en la apropiación para que exista una mejor correlación en el desarrollo curricular de la asignatura “matemáticas discretas”.

La presente investigación aporta teóricamente en cuanto a la relación de la matemática discreta con los temas asociados de la computabilidad; desde lo práctico al desarrollo y construcción de una máquina de Turing, por parte de los estudiantes; para con ello apropiar el concepto computable con la intencionalidad que la comunidad académica comprenda la "matemática discreta" como una herramienta hacia la construcción de una matemática aplicada, complementando su formación integral como ingenieros o ingenieras en su formación tecnológica.

Realizar este trabajo de grado es una responsabilidad para la formación disciplinar de la matemática en la ingeniería de sistemas y así poder ser partícipes del desarrollo científico-técnico en la ingeniería de sistemas.

1.4 Objetivos

1.4.1 Objetivo general

Analizar la relación entre los niveles de apropiación del concepto de "computabilidad" y las dificultades que experimentan los estudiantes del curso de «matemáticas discretas» de ingeniería de sistemas en la Escuela Tecnológica Instituto Técnico Central – ETITC.

1.4.2 Objetivos específicos

- Determinar los niveles de apropiación del concepto de "computabilidad" en los estudiantes del curso "matemáticas discretas" del programa de Ingeniería de Sistemas.
- Describir las dificultades asociadas a la apropiación del concepto de "computabilidad" de los estudiantes del curso "Matemáticas discretas" del programa de Ingeniería de Sistemas.
- Plantear una serie de lineamientos que orienten la enseñanza de la "computabilidad", a estudiantes del programa de Ingeniería de Sistemas.

1.5 Supuestos de investigación

La computabilidad se visualiza más desde el punto de vista práctico, de su funcionalidad, de su aplicabilidad; por ejemplo, para los 4.540 millones de usuarios de Internet los cuales representan el 59% de la población mundial, la automatización ha logrado que los procesos en línea sean sencillos y transparentes, esa es la visión de usuario final en lo que se ha dado en llamar una era digital.

Las matemáticas son el fundamento detrás del desarrollo tecnológico y es lo invariante: el álgebra booleana, el concepto de función y la lógica matemática permanecen tal como fueron formalizadas en el siglo XIX; mientras los lenguajes de programación han tenido varios paradigmas en los últimos 50 años tales como programación imperativa, programación estructurada, programación orientada a objetos o programación funcional.

El desarrollo de la investigación pretende identificar dónde están las dificultades en la apropiación del concepto de computabilidad.

1.6 Delimitación y limitaciones

1.6.1 Delimitación

El trabajo de campo de este proyecto realizó en la Escuela Tecnológica Instituto Técnico Central (ETITC). con domicilio principal en la Calle 13 #16-74 de la ciudad de Bogotá, Distrito Capital de Colombia.

Actualmente está posicionada como la primera Escuela Tecnológica oficial de Colombia y ofrece programas de Técnico Profesional, Tecnólogo, Profesional en Ingenierías, Especializaciones, Diplomados, Educación a Distancia y Educación Continuada.

En particular se desarrollará en el programa curricular de Ingeniería de Sistemas con la materia “Matemáticas discretas”, la cual corresponde al segundo semestre de los once semestres de la malla curricular. Cada grupo se compone en promedio de 15 alumnos, con edades comprendidas entre los 18 y los 22 años.

Las matemáticas discretas incluyen entre otros los siguientes temas: lógica, álgebra de Boole, conjuntos, teoría de grafos, árboles, funciones y relaciones. Las mayores dificultades de los estudiantes están asociadas con la comprensión de conceptos matemáticos de la lógica, la algebrización y la operabilidad de la formalización.

Los resultados esperados es que los estudiantes apropien el concepto de computabilidad el cual es importante para la formación académica y profesional del ingeniero de sistemas en general y para materias como: programación de computadores, base de datos y estructuras de información entre otras. El trabajo de campo se realizó a lo largo de un semestre, el cual correspondió al primer semestre del año 2020.

1.6.2 Limitaciones

- La principal limitación tiene que ver con el alto nivel de complejidad del concepto de "computabilidad", el cual no termina de construirse y se encuentra en constante desarrollo.
- El tiempo de la asignatura Matemáticas Discretas es de dos (2) horas semanales, el total de semanas de duración del semestre son 16, para un total de 32 horas semestrales. En los meses de diciembre, enero, junio y julio no hay interacción con la población en estudio.
- Existen pocos estudios sobre la enseñanza del tema de computabilidad desde una perspectiva diferente a la enseñanza de "programación de computadores", que es la manera como los estudiantes aprenden "Lenguajes de programación".
- Esta es la primera vez que en la ETITC se realizó un estudio de estas características en la materia "matemáticas discretas", es importante poder darle continuidad a este tipo de proyectos.
- La crisis generada por el covid19 obligó a la realización de clases de forma remota, es la nueva realidad en la forma de enseñanza.

1.7 Definición de términos

Computabilidad: técnicamente se define como la formalización del concepto intuitivo de algoritmo. Es el estudio de los modelos relacionados con los problemas de decisión que se pueden resolver con un algoritmo o equivalentemente con una máquina de Turing, porque todos los problemas no necesariamente pueden ser resueltos.

Didáctica: etimológicamente viene del griego "didastékene", que significa enseñar y tékene significa arte; arte de enseñar. La didáctica es la rama de la pedagogía que busca métodos, establece pautas y técnicas para mejorar la enseñanza.

Función Computable: es uno de los objetos básicos de estudio en la teoría de la computabilidad y son, específicamente, las funciones que pueden ser calculadas por una máquina de Turing.

Máquina de Turing: es un dispositivo que manipula símbolos sobre una tira de cinta de acuerdo con una tabla de reglas. A pesar de su simplicidad, una máquina de Turing puede ser adaptada para simular la lógica de cualquier algoritmo de computador y es particularmente útil en la explicación de las funciones dentro de un computador.

Matemáticas Discretas: es una rama de las matemáticas encargadas del estudio de los conjuntos discretos: finitos o infinitos numerables. La matemática discreta es la base de la matemática computacional. Mientras la matemática continua se encarga del estudio de conceptos como la continuidad y el cambio; la matemática discreta estudia estructuras cuyos elementos pueden contarse uno por uno separadamente.

Número Computable: es un número real que puede ser aproximado por un algoritmo con un nivel de exactitud arbitrario.

Capítulo 2. Marco Referencial

El marco de referencia para la realización de este trabajo es la "matemática discreta", con una mirada global desde las diferentes disciplinas que estructuran la computabilidad, abordando las siguientes preguntas: ¿cuál es el concepto de computabilidad que se va a utilizar en este trabajo?; de hecho, existen muchas definiciones. ¿Cuál es su importancia?, en el momento actual. ¿Cuáles son sus fundamentos? y ¿cuál es su didáctica?

La materia "matemáticas discretas" está al principio del plan de estudios de ingeniería de sistemas, pero lo más importante es hacer de estos contenidos una de las primeras aproximaciones del estudiante al concepto de computabilidad, conceptualización que debe ser sencilla, clara y entendible para poder tener una panorámica que le acompañe a lo largo de su formación académica y luego en su vida laboral.

Se hace una pregunta alrededor del desarrollo curricular de la "matemática discreta". Los contenidos del ETITC, se comparan con la Universidad Nacional de Colombia, y a nivel internacional se toma como referente el Massachusetts Institute of Tecnology –MIT-.

Se referencian los tres libros de texto más reconocidos en la materia, los cuales sirven de guía para la enseñanza de la "matemática discreta" son: (1) Kenneth H. Rosen, (2) Richard Johnsonbaugh, y (3) de la Serie Schaum, cuyos autores, Seymour Lipschutz y Marc Lipson; se puede concluir sin lugar a dudas, que existe un consenso internacional en las temáticas.

Actualmente, el concepto de computabilidad se estructura a partir de las cinco propuestas más reconocidas en la década del 30 del siglo XX, las cuales son: (1) Máquina de Turing, (2) Cálculo Lambda, (3) Funciones Recursivas, (4) Máquinas de Post y (5) Algoritmos de Markov.

Han transcurrido ocho décadas, este tiempo es un período suficiente para afirmar que la propuesta de Alan Turing es la más reconocida y es un argumento de este trabajo; y la didáctica es el elemento diferenciador; pero aún más importante, el modelo de Turing no ha sido superado.

Al tratar de entender la computabilidad desde cada una de las siguientes seis ramas que forman la matemática discreta: lógica, teoría de conjuntos, teoría de números, álgebra booleana, grafos y árboles; se espera tener como resultado una mayor comprensión y entendimiento de la materia, justificar al estudiante la necesidad de adentrarse en la profunda y difícil formación matemática que se requiere y la necesidad de la apropiación del concepto de la computabilidad para su formación disciplinar en ingeniería de sistemas.

2.1 Computabilidad

2.1.1 Definición de computabilidad

Todos de una u otra manera comprenden el significado de "computabilidad", porque es la realidad avasalladora de los 5.000 millones de dispositivos que están utilizándose hoy en día, 65% de la población mundial según cifras conservadoras; pero existe una enorme diferencia entre las personas del común y quienes van a ser los hacedores científico-técnicos de esta realidad, los jóvenes que se preparan en esta rama del conocimiento, por ello es necesario identificar las dificultades para la apropiación de dicho concepto.

Una de las premisas de este trabajo de investigación: "La computabilidad es un resultado de la modernidad", pero no es nueva. El hombre prehistórico ya computaba los miembros de su familia y en sus actividades cotidianas. En la milenaria cultura china, el ábaco era el computador de sus tiempos; en la antigua babilonia hace ya 3.800 años, las tablillas de arcilla eran los ordenadores de sus tiempos, además allí escribieron algoritmos cuneiformes para la resolución de ecuaciones de segundo grado; la matemática griega dejo un enorme legado en este sentido y los hindúes aportaron el sistema de numeración decimal.

En la edad media los árabes pusieron su impronta con la palabra “algoritmo” y el álgebra en el siglo IX; en la península ibérica Ramón Lull (1236-1316) ideó un método general de resolución de problemas que llamó “ars magna”; Leibniz (1646-1716) en lo que es hoy Alemania propone el “calculus ratiocinator”; George Boole (1815-1864) escribe los fundamentos de la aritmética computacional moderna; Gottlob Frege (1848-1925) hace el cálculo cuantificacional; Hilbert (1862-1943) propone su proyecto de formalización y se tiene por fin el siglo XX donde se logra la concretización de la quimera computacional.

Este proceso no es un desarrollo puntual y menos lineal, debe entenderse como el acumulado histórico de la cultura humana y la historia en este caso ofrece puntos donde detenerse para un mejor entender el presente, ese punto es el año 1936 con el paper elaborado por Alan Turing y esa es la definición que se toma en este trabajo.

Tal como se expresa en el libro: «Computability. Turing, Godel, Church, and Beyond», editado por B. Jack Copeland, Carl J. Posy, and Oron Shagrir: “Se puede decir que la revolución computacional de la década de 1930 impulsó la expansión vertiginosa de los horizontes de conocimiento que caracterizan nuestra era científica moderna” (p. 4).

2.1.1.1 Números computables

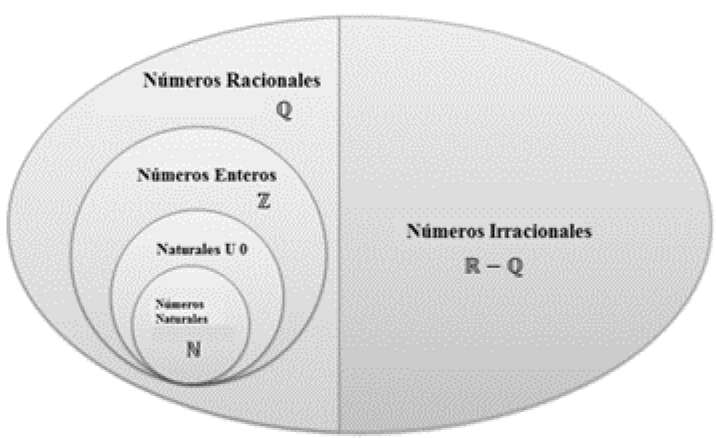
La computabilidad es la caracterización de una propiedad específica de “ser” o “no ser”, y es computable si es resoluble siguiendo un algoritmo específico el cual entrega una solución en un número determinado de pasos.

El conjunto de los números pares es un conjunto computable, al igual que el conjunto de números primos; mientras el conjunto de los polinomios con coeficientes enteros, y en varias variables, que tienen raíces enteras no es computable.

La computabilidad, también puede ser entendida de abajo hacia arriba (bottom - up) comenzando en objetos computables y objetos no computables, números computables y números no computables, conjuntos computables y conjuntos no computables, funciones computables y funciones no computables; hasta llegar a su propio límite teórico de clasificar los problemas en computables e incomputables.

Con la idea de "números computables" comienza el famoso paper de Alan Turing, es más, este concepto hace parte del mismo título del paper: "On computable numbers …", de manera textual Turing define los números computables como "los números reales cuya expresión digital es calculable con medios finitos"; los cuales en forma general se establecen desde la definición de números reales, la estructura general de los números reales es la siguiente.

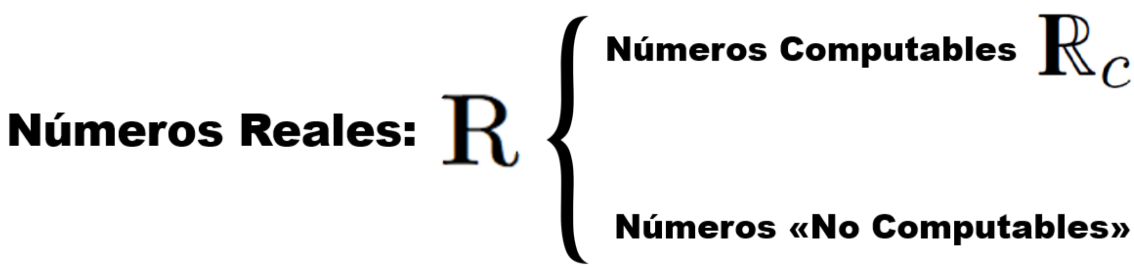
Figura 4: Estructura de los Números Reales

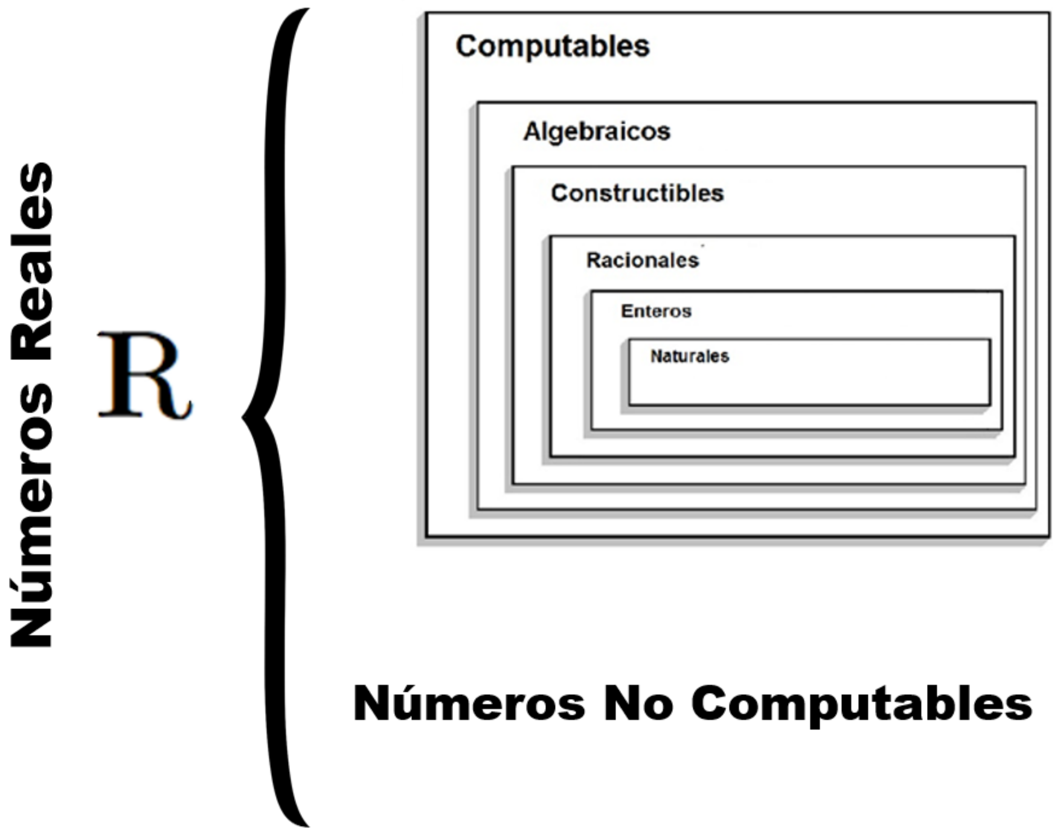


Fuente: Dedekind, R. (1872) Continuity and irrational numbers.

Uno de los primeros en dar una definición formal de los números reales fue el matemático Richard Dedekind hacia el año de 1872. Desde la definición de "número reales" se estructura la definición de "números computables"; al dividir el conjunto de los reales en números computables y números no computables; lo que hace Turing es trasladar la definición de números reales que tanto había costado en la matemática a la nueva entelequia que él se inventó: los "números computables", y dividió dicha abstracción al igual que los números reales cuando se dividen en números racionales y números irracionales. Los números reales no son un conjunto numerable, pero un subconjunto de los reales como los números computables son los únicos números reales que podemos calcular, poder computar es una propiedad es muy importante para entender la nueva categoría matemática que se equipara a los átomos en la estructura de la materia. Para Turing, los "números computables" son los átomos de la computabilidad.

Figura 5. Clasificación de los números reales en función de la computabilidad.





Fuente: construcción de la autora

Los números computables son los números reales que pueden ser computados con la precisión que se desee por un algoritmo. Los números computables además tienen la propiedad de ser infinitos, lo que significa que se requiere un espacio y un tiempo infinito para poder enumerarlos a todos. Siguiendo la comparación con los números reales, los números reales son innumerables lo que significa que aun teniendo un espacio y un tiempo infinito no podríamos enumerarlos uno a uno. Resumiendo, si existe un algoritmo capaz de calcular todos los dígitos que caracterizan un número, este es un número computable.

Se sabe que las constantes fundamentales como π, $\sqrt{2}$, e y los números algebraicos irracionales son todos computables.

Un número real es computable si existe una función computable que calcula cada uno de sus dígitos fraccionarios. Es decir, si existe $f: N \rightarrow N$ computable tal que para todo $n, f(n)$ devuelve el n-ésimo dígito fraccionario del número expresado en cierta base.

$$\boldsymbol{e = \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^{n}}$$

El número $\boldsymbol{e}$ es un número computable, también es un número trascendente e irracional, pero $\boldsymbol{e}$ no es un número algebraico. Se tiene la forma de calcular el número con toda precisión que deseemos, basta elegir n suficientemente grande. Es decir, se tiene un "algoritmo" que permite saber con precisión la n-ésima cifra de la expansión decimal de $\boldsymbol{e}$.

Definición formal y moderna de *número computable.*

Un número real $\boldsymbol{a}$, es computable si se puede dar una aproximación de él mediante una función computable de la siguiente forma:

Dado cualquier número entero $\boldsymbol{n \geq 1,}$

la función produce un número entero $\boldsymbol{k}$ tal que:

$$\boldsymbol{\frac{k-1}{n} \leq \ a \ \leq \frac{k+1}{n}}$$

En el siglo XIX la matemática sufrió una de las más grandes revoluciones, el concepto de número, el de función y el de conjunto; estos conceptos fueron fundamentales para la construcción del concepto de “computabilidad”.

> Esta corriente se autodenomina “clásica”, pero la llamaré “conjuntista” porque coloca en el centro de la matemática, en una forma u otra, la noción de conjunto y trabaja en fortalecerla. Iniciada por Dedekind (1831-1916) y Cantor (1845-1918), incorpora logros de Frege (1848-1925), Peano (1858-

> 1932), Whitehead (1861-1947) y Russell (1872-1970), y recibe aportes de Hilbert (1862-1943), Zermelo (1871-1953), Tarski (1902-1983), von Neumann (1903-1957), Gödel (1906-1978), Gentzen (1909-1945), y muchos otros. Por otra parte, están los adversarios del conjuntismo ilustres matemáticos como Kronecker (1823-1891), Poincaré (1854-1912), Brouwer (1881-1955) y Weyl (1885-1955), filósofos como Wittgenstein (1889-1951) y Lorenzen (1915-1994) que impugnan con poderosas razones sus ideas y prácticas más arraigadas, sin que la masa de los matemáticos les preste mucha atención (Torretti, 1998, p. 11).

Además del concepto de conjunto, que es base para el concepto de función, se encuentra otro objeto básico en la teoría de la computabilidad que es el concepto de función computable, y son aquellas funciones que pueden ser calculadas por una máquina de Turing.

Ejemplos de funciones computables son la suma y la multiplicación y para más precisión, toda función con dominio finito es efectivamente computable:

El concepto de función también sirve para estructurar el concepto de número computable definido desde el concepto de función y específicamente desde el concepto de funciones recursivas.

Relación entre función computable y número computable

Un número real es computable si existe una función computable que calcula cada uno de sus dígitos fraccionarios. Es decir, si existe $f: N \rightarrow N$ computable tal que para todo $n, f(n)$ devuelve el n-ésimo dígito fraccionario del número expresado en cierta base.

Representación del número computable α

Dado i tal que $\varphi_i: \mathbb{N} \rightarrow \{0,1\}$ es una función total recursiva, consideramos el número real computable α

$$\alpha = \sum_{n=0}^{\infty} \frac{\varphi_i(n)}{2^n}$$

“Un objeto es computable si es computable por una máquina de Turing”, esta es la definición de computabilidad que se asume en este trabajo.

> A computation is a process whereby we proceed from initially given objects, called inputs, according to a fixed set of rules, called a program, procedure, or algorithm, through a series of steps and arrive at the end of these steps with a final result, called the output. The algorithm, as a set of rules proceeding from inputs to outputs, must be precise and definite, with each successive step clearly determined. The concept of computability concerns those objects which may be specified in principle by computation. (Soare, 1996, p.78).

En computabilidad, el concepto de lo numerable es fundamental, porque solo son computables las funciones de conjuntos numerables; y todo lo que es computable es expresable, expresable en máquinas de Turing.

Se concluye entonces, que las matemáticas proporcionan el fundamento teórico para el estudio de la computabilidad.

2.1.2 Desarrollo histórico de la computabilidad

Lo computable ha estado presente en la cultura humana, es una premisa para el desarrollo de este trabajo de investigación. El homo sapiens ya logra diferenciar uno del otro, puede saber cuántos son los miembros de su familia, la dimensión del tiempo que es un continuum se puede discretizar en el número de días o el número de noches, la astronomía que es de las más antiguas ciencias arroja luces en cuanto a la discretización, porque los objetos celestes observables en la noche son discretos; los dedos de una mano son cinco, además con los dedos de la otra mano suman 10, por esta razón la base numérica más difundida en el mundo, es el sistema decimal y su fácil operabilidad hace que termine imponiéndose sobre el sistema de numeración romano, maya u otro cualquiera.

Un ejemplo de la falta de contexto histórico en el tratamiento de lo computable se puede apreciar en muchos de los textos sobre “matemática discreta”, donde ubican los comienzos de esta disciplina en la década de 1930, en contraste con la idea de este trabajo, el cual considera la existencia de la computabilidad desde los albores mismos de nuestra especie y está estrechamente relacionado con las matemáticas, las cuales han estado presente en la historia de la humanidad y en todas las culturas para la solución de los problemas en el día a día.

El ser humano primero hizo uso de la computabilidad antes de la invención de la escritura. Shurkin (1984) expresa: “El hueso de Ishango, data del paleolítico superior 20.000 a.n.e. tiene muescas donde queda la huella de lo enumerable y lo contable” (p.21)

Otro ejemplo, es el ábaco, instrumento que sirve para computar, el cual hizo parte de la antigua cultura china.

El algoritmo de Euclides para encontrar el máximo común divisor de dos números es el mejor ejemplo de los pasos precisos, con un principio y un fin, un input y un output.

En la cultura griega la silogística de Aristóteles (384 - 322), la geometría de Euclides (325 - 265) y la mecánica de Arquímedes (287 - 212) ya estaba presente el método axiomático, porque esta es otra forma de plantear la pregunta de Hilbert: ¿es posible resolver todos los problemas matemáticos de forma axiomática?

Lo que dejó la década del 30 del siglo XX fue varias propuestas de matematización del concepto de algoritmo: la recursividad, el operador Lambda, la máquina de Turing, las máquinas de Post y los algoritmos de Markov como las propuestas más relevantes.

En particular, la propuesta de Alonzo Church (operador Lambda) y la de Alan Turing (Máquinas de Turing) se hicieron una independiente de la otra; el trabajo de Church es publicado con unos pocos meses de anterioridad al trabajo de Turing. En Estados Unidos Alonzo Church era profesor del Instituto de altos estudios de la Universidad de Princeton IEA por sus siglas en inglés; mientras Turing al otro lado del Atlántico en la Universidad de Cambridge donde fue estudiante de pregrado en matemáticas y también profesor.

Todos son matemáticos de formación que trabajaban para esta época en la lógica, la fundamentación de la matemática y el análisis funcional.

No era predecible que dichas investigaciones conformarían la base de la era computacional, no era eso lo que buscaban, estaban interesados en los fundamentos de la matemática, la computabilidad se la encontraron en el camino.

Hilbert, uno de los matemáticos más importantes entre el fin del siglo XIX y el siglo XX, pretendía crear un sistema matemático formal completo y consistente en el cual todas las aseveraciones matemáticas fueran planteadas con precisión; el proyecto de Hilbert era el método formal axiomático y dentro de esta línea buscaba un procedimiento algorítmico general para resolver problemas matemáticos.

2.1.3 La computabilidad de las máquinas de estado finito.

Una máquina de estado finito es un Autómata Finito Determinístico y se define por lo que corresponde a cada una de las tres palabras:

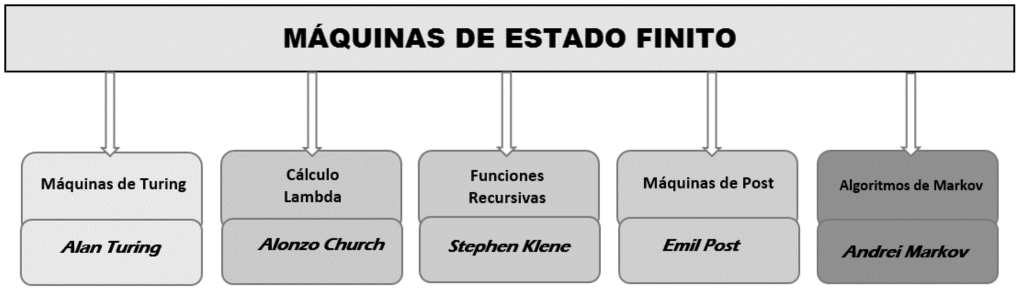
1. **Autómata** porque sirve para describir sistemas automáticos.

2. **Finito** porque tiene un número finito de estados, y

3. **Determinístico** porque en cada estado sólo hay una forma de ir a otro estado dependiendo de la entrada que llega.

Para efectos de este trabajo, se estudiaron los cinco siguientes modelos.

Figura 6. Máquinas de Estado Finito



Fuente: construcción de la autora

Estos cinco modelos son equivalentes y es importante recalcar que todos son abstracciones matemáticas, son intentos de capturar la computabilidad, de matematizarla.

2.1.3.1 Máquina de Turing.

La máquina de Turing se describe completamente en el paper de 36 páginas, escrito en 1936, denominado "Los números computables, con una aplicación al Entscheidungsproblem".

El primer párrafo lo dedica exclusivamente a los números computables y el título del mismo paper: "Sobre números computables…"; comienza con pies firmes, la teoría de números es una de las ramas más antiguas y desarrolladas de la matemática, Turing comienza en bases muy sólidas a construir su catedral. Analiza la noción intuitiva de 'algoritmo', dando lugar a la definición de máquina de Turing, la cual es básicamente un modelo matemático que se puede visualizar en un dispositivo.

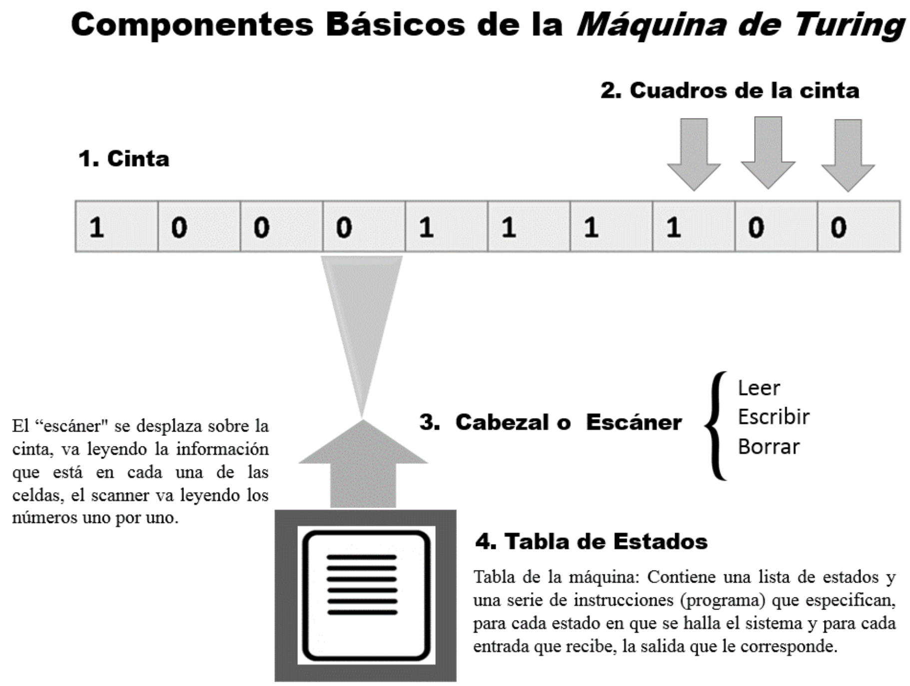
La esencia de la filosofía de Turing es mecanicista y así lo definió en su paper de 1936. Es importante reiterar que la máquina de Turing, es una máquina abstracta que corresponde a la noción del concepto de algoritmo, esta máquina como la concibió Turing no es un objeto físico, es una formalización de una máquina que manipula símbolos sobre una cinta de acuerdo a una tabla de reglas que trabaja de manera secuencial paso a paso.

El propósito de Turing al describir su máquina era reducir los cálculos a sus rasgos esenciales más simples, describiendo de un modo sencillo algunos procedimientos básicos que son manifiestamente efectivos y a los que pueda reducirse cualquier procedimiento efectivo.

La máquina tiene un input y un output; una entrada y una salida; es la misma definición de función que fue una de las más importantes elaboraciones matemáticas del siglo XIX. La máquina recibe la información de entrada desde el "papel" y al final escribe la respuesta, el cálculo de la función en ese mismo "papel", en esa misma cinta.

La máquina ideada por Turing tiene una cinta infinita en ambas direcciones la cual se divide en cuadros y cada cuadro recibe el nombre de celda, la cual está en blanco o tiene un símbolo, una sola y única marca. Estas celdas hacen discreto el dispositivo porque se puede distinguir uno del otro.

Figura 7. Máquina de Turing



Fuente: construcción de la autora

La cinta está formada por cuadros, llamados también celdas. Cada celda de la cinta está en blanco o contiene una sola y única marca y en un momento determinado la máquina puede leer un símbolo el cual puede permanecer igual, cambiarse o ser borrado. En la descripción de la cinta está el concepto de lo que era una máquina de escribir de aquellos tiempos. El trabajo de Turing se basa en una caracterización de lo que en su época se llamaba literalmente un agente computacional humano; aquí se ve al Turing de sus tiempos, esta era la actividad que desempeñaban personas reales de carne y hueso, haciendo largos cálculos los cuales debieran ser hechos ahora por una máquina, de manera intuitiva es la representación de una función producida por un procedimiento mecánico. Se refiere a algo abstracto, que nunca va a ser materializado en el sentido que ese agente ideal de Turing se concibe como un individuo con papel y lápiz de manera infinita lo cual nunca puede ser real, porque en esencia todo es un modelo; mientras que los símbolos y los estados si tienen el carácter de finitud.

El procedimiento del cálculo del valor de una función de ese agente ideal se lleva a cabo escribiendo los símbolos en un papel, estructurado inicialmente en forma unidimensional y dividido en celdas. El agente, que en una máquina sería el escáner o la cabeza lectora, esta parte del dispositivo es capaz de observar, a la vez sólo un número finito de estos cuadrados, uno a la vez y también puede efectuar un número finito de operaciones atómicas o elementales.

Éstas son operaciones tan simples que no sería fácil imaginárselas subdivididas en otras operaciones, y su ejercicio dependerá de los estados mentales del agente ideal, así como de los símbolos escritos en los cuadrados observados.

Las operaciones atómicas constituyen cambios de símbolos en los cuadrados observados, movimientos a diversos cuadrados observados, pero a cierta distancia del cuadrado observado por el agente ideal. Turing también impuso la condición determinista de que a lo sumo se podía efectuar una operación atómica a partir de un estado mental y un símbolo observado.

Luego de que Turing caracteriza al agente ideal y a lo que se ha dado en llamar "máquinas de Turing", prueba de manera rigurosa la siguiente proposición: toda función calculable por el agente ideal es calculable por una máquina de Turing. De aquí procede a aseverar que toda función computable en forma efectiva es computable por el agente ideal y, por lo tanto, es computable por una máquina de Turing, esto es:

Tesis de Turing: Toda función computable es calculable por una máquina de Turing.

Una máquina de Turing puede ser definida como una séptupla:

$$\mathbf{M = (Q, \Sigma, \Gamma, s, b, F, \delta)}$$

Q: Es un conjunto finito de estados $\mathbf{q_1, q_2, q_3, \ldots}$

Σ: Es un conjunto finito de símbolos distinto del espacio en blanco, denominado alfabeto de máquina o de entrada.

Γ: Es un conjunto finito de símbolos de cinta, denominado alfabeto de cinta.

s: Es el estado inicial.

b ∈ **Γ:** Es un símbolo denominado blanco, y es el único símbolo que se puede repetir un número infinito de veces.

F ⊆ **Q** Es el conjunto de estados finales de aceptación.

δ: Q x **Γ** → **Q** x **Γ** x **{ L** x **R}** Es una función parcial, denominada función de transición, donde L es un movimiento a la izquierda y **R** es un movimiento a la derecha.

La importancia del paper de Turing la reconoce Stephen Hawking (2006) cuando escoge este artículo como el último de los 31 grandes logros matemáticos de la historia de la humanidad en su libro: "Dios creó los números. Los descubrimientos matemáticos que cambiaron la historia."

En la práctica docente del ETITC este documento se propone como la guía para el desarrollo de la "Máquina de Turing", tarea que cada alumno debe hacer a lo largo del semestre; los resultados han sido muy satisfactorios por el compromiso que asumen los estudiantes y la importancia de esta temática en su formación profesional.

2.1.3.2 Cálculo lambda.

El cálculo lambda (λ) es un formalismo matemático que fue introducido por Alonzo Church en el año 1934. Se define ***f*** como una función calculable en forma efectiva, si para todo entero positivo ***a***, existe un entero positivo ***b***, tal que ***f (a) = b.*** Esto es un teorema que puede ser probado.

El premio nobel de física año 2020, define el cálculo lambda resumidamente.

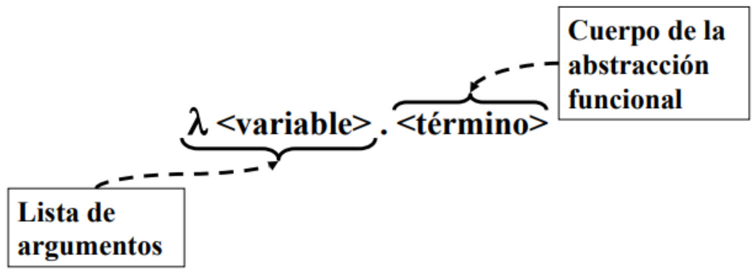
> Los símbolos del formalismo son el operador **λ**, los tres pares de paréntesis: ( ), [ ],{ }, y una lista de variables ***a***, ***b***, ***c***, …***z, a'***, ***b'***, ***c'*****, …,** ***z'***, ***a''***, ***b''*****, …** ***a'''*****, …** ***a''''*****, …** cantidad ilimitada de símbolos, cada uno de los cuales representa una función o más precisamente una "operación matemática". Los "argumentos" de estas funciones, es decir, las cosas sobre las que estas funciones actúan, son otras

cosas del mismo tipo, esto es, también funciones. Además, el resultado (o “valor”) de una función actuando sobre otra es de nuevo una función. (Penrose, 1989, p. 69).

En el cálculo Lambda el centro de importancia es la recursividad.

El cálculo Lambda presenta una nueva sintaxis, la abstracción funcional es:

Figura 8. Abstracción funcional del Cálculo Lambda.



Fuente: gemmatmtz

La sintaxis propuesta utiliza la representación de funciones con un solo argumento.

En caso de que la función requiera más de un argumento se representa de la siguiente forma:

$$\lambda x_1 \cdot \lambda x_2 \dots \lambda x_n \cdot M$$

En el cálculo lambda una expresión o término se define recursivamente a través de las siguientes reglas de formación:

1. Toda variable es un término: x, y, z, u, v, w, x_1, x_2, y_9, ...
2. Si ***t*** es un término y ***x*** es una variable, entonces (**λ*x*.*t***) es un término (llamado una abstracción lambda).
3. Si ***t*** y ***s*** son términos, entonces (***ts***) es un término (llamado una aplicación lambda).
4. Nada más es un término.

En el modelo de Alonzo Church, el **0**, **1** y **2** se definen así:

0	se representa por:	**λ*ab*.*a*(*b*)**
1	se representa por:	**λ*ab*.*a*(*a*(*b*))**
2	se representa por:	**λ*ab*.*a*(*a*(*a*(*b*)))**

2.1.3.3 Funciones recursivas.

El concepto de funciones recursivas proviene del desarrollo matemático del siglo XIX, Kurt Gödel resolviendo el teorema de incompletitud se refiere a este concepto, Alonzo Church y Stephen Kleene hicieron grandes aportes; lo que es claro en este tipo de temas es el alto nivel de formalización.

Una función $\boldsymbol{\phi}(\boldsymbol{x_1}, \ldots \boldsymbol{x_n})$ es recursiva parcial si existe un sistema $\boldsymbol{E}$ de ecuaciones que la define recursivamente en el sentido que se explica a continuación. Un sistemas $\boldsymbol{E}$ de ecuaciones define recursivamente una función recursiva parcial n-aria si, para cada n-tupla $\langle \boldsymbol{x_1}, \ldots \boldsymbol{x_n} \rangle$ de números naturales se puede derivar de $\boldsymbol{E}$, conforme a las reglas de derivación, a lo sumo para un numeral $\boldsymbol{X}$ una ecuación de la forma $\boldsymbol{f}(\boldsymbol{X_1}, \ldots, \boldsymbol{X_n}) = \boldsymbol{X},$ donde $\boldsymbol{X_{1,\ldots,}X_n}$ son los numerales que representan a los números naturales $\boldsymbol{x_{1,\ldots,}x_n}$. La función n-aria definida por $\boldsymbol{E}$ en este caso es la función $\boldsymbol{\phi}$ cuyo valor $\boldsymbol{\phi}(\boldsymbol{x_1}, \ldots, \boldsymbol{x_n})$ para el argumento $\langle \boldsymbol{x_1}, \ldots, \boldsymbol{x_n} \rangle$ es el número natural $\boldsymbol{x}$ representado por el numeral $\boldsymbol{X}$ se ese numeral existe, y de otro modo esta indefinido.

Se fijará una enumeracion calculable $\{M_e\}$, $\boldsymbol{e} \in \mathbb{N}$, de las maquinas de Turing y se denota por $\varphi_e^{(m)}$ la funcion recursiva parcial de $\boldsymbol{m}$ variables computada por la maquina de Turing, $\boldsymbol{M_e}$.

Tenemos el conocido teorema ***s-m-n*** de Kleene.

Teorema. Para todo *m, n, k* ≥ 1, existe una funcion recursiva total s_n^m: $\mathbb{N}$ x $\mathbb{N}^m \rightarrow \mathbb{N}$ tal que para todo **e** ϵ $\mathbb{N}$ si $\varphi_e^{(n+m)}$:$\mathbb{N}^{n+m} \rightarrow \mathbb{N}^k$ es recursiva parcial, entonces para todo v$\in \mathbb{N}^n$ se cumple que

$$\varphi_{s(e,v)}^{n}(u) = \varphi_e^{n+m}(v, u), \quad \text{para todo } \mathbf{u} \in \mathbb{N}^n.$$

Recordemos que $\boldsymbol{A} \subseteq \mathbb{N}^m$ es recursivamente enumerable.
Si $\boldsymbol{A}$= dom(φ), con φ recursiva parcial y que $\boldsymbol{A}$ es recursivo si tanto $\boldsymbol{A}$ como $\boldsymbol{A^c}$ son recursivamente enumerables (o equivalentemente si su correspondiente función caracteristica

χ_{A}: $\mathbb{N}^m \rightarrow \{0,1\}$ es recursiva.

El ejemplo más célebre de un conjunto recursivamente enumerable que no es recursivo es el conjunto de parada:

$$\mathbf{K}= \{\eta | \eta \ \in \mathrm{dom}(\varphi_n)\},$$

Cuya importancia es vital en la teoría de la recursión.

2.2 Matemática Discreta

2.2.1 Definición de Matemática Discreta.

La "matemática discreta" estudia los elementos discretos, la relaciones entre ellos y sus operaciones. La palabra «discreta» según el diccionario de la RAE (2019) es un adjetivo que concuerda con el nombre o pronombre que califica en género y número, «discreto/masculino», «discreta/femenino». "Discreto" proviene del latín [dıˇscre- tus] y es el participio de discernere, discernir, «separar una cosa de otra».

El significado de lo discreto, desde el punto de vista matemático está relacionado con el conteo de los objetos, porque cuando es posible diferenciar un elemento del otro, se pueden contar uno por uno separadamente mediante procesos finitos, numerables y contables. La sucesión de los números enteros es discreta, pero el volumen de agua en un recipiente es continuo.

La "matemática discreta" es la ciencia que trata sobre el conocimiento y explicación de fenómenos discretos y procesos finitos y vista de forma general pareciera que abarcara una gran variedad de temas sin conexión: lógica, teoría de conjuntos, álgebra de Boole, funciones, recursividad, árboles y grafos; pero si se observa desde el punto de vista de las estructuras es la rama de la matemática que estudia las estructuras discretas que son estructuras abstractas las cuales se usan para representar objetos discretos que pueden ser finitos o infinitos. Ejemplos de estructuras discretas son: los conjuntos, las permutaciones, las relaciones, los grafos, los árboles y las máquinas de estado finito.

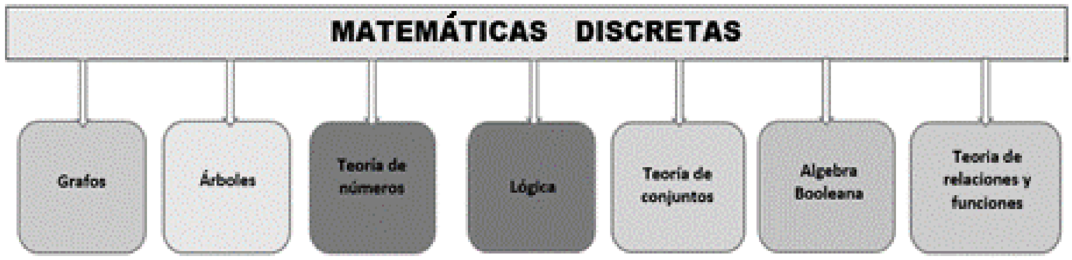
Otra manera de hacer una aproximación a la matemática discreta, es aquella rama de la matemática que trata del estudio de conjuntos inconexos, donde no existen conexiones topológicas entre los elementos tales como los números enteros, los grafos y las sentencias lógicas (proposiciones).

La matemática continua se diferencia de la matemática discreta en que estudia conjuntos no numerables, fenómenos continuos los cuales explicaron de manera técnico-científica los procesos desencadenantes de la revolución industrial que comenzó en el siglo XVII, porque primero fue el invento y luego la explicación; en contraste en esta nueva era, primero es el conocimiento antes que el invento y en este invento es la matemática discreta la base del conocimiento científico que subyace en la era computacional.

Existen otras diferencias entre los dos tipos de matemática. En cuanto a la cantidad; la cantidad continua es mensurable, la cantidad discreta es numerable y bajo la lupa de la computabilidad se aritmetiza el análisis porque “solo son computables” las funciones de conjuntos numerables.

La matemática discreta unifica desde el complejo y extenso universo matemático lo que se requiere para el desarrollo de la computabilidad.

Figura 9. Ramas de la Matemática Discreta.



Fuente: construcción propia.

2.2.2 Importancia de la matemática discreta.

La matemática discreta es el fundamento del desarrollo computacional y surge como una rama que engloba varias disciplinas: teoría de conjuntos, funciones y relaciones, lógica proposicional, álgebra booleana, combinatoria, autómatas finitos, grafos, árboles y criptografía.

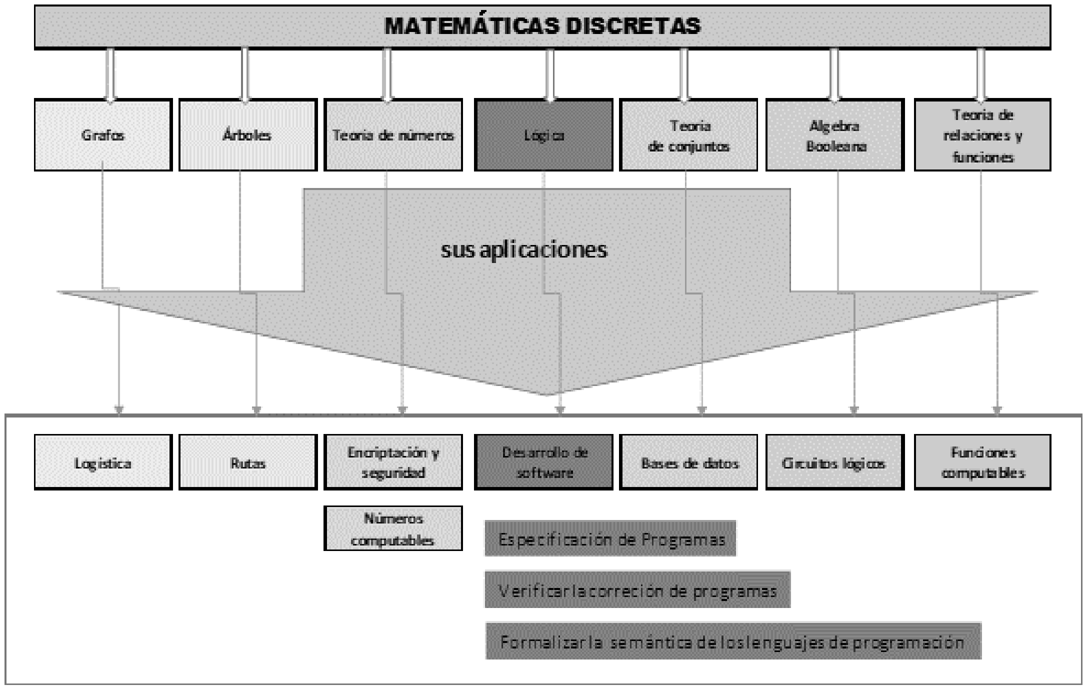
La matemática en los últimos 300 años ha dado mayor importancia a la matemática continua que estudia las estructuras del movimiento, en especial durante el siglo XVIII con los trabajos del cálculo de Newton y Leibniz.

> Las recomendaciones curriculares recientes del Instituto de Ingenieros Eléctricos y Electrónicos de la Computer Society (IEEE-CS) y la Association for Computing Machinery (ACM) incluyen las matemáticas discretas como la mayor parte del "conocimiento básico" para estudiantes de computación quienes deben tomar al menos un curso de un semestre en el tema como parte de sus estudios de primer año y preferiblemente dos semestres cuando sea posible. Este libro incluye los temas recomendados por esos organismos. (Epp, 2011, p. 12).

Se debe estudiar y comprender a fondo la matemática discreta porque son el fundamento de la computabilidad.

Algunas de las aplicaciones de cada una de las ramas de la matemática discreta tal como se muestra en la gráfica.

Figura 10. Aplicaciones de la matemática discreta.



Fuente: construcción propia.

No se ha completado aún el primer siglo del proceso de formalización de los años 30, pero los desarrollos son inigualables en el siglo XXI; dichos inventos se popularizaron en las tres últimas décadas y ello hace pensar a algunos que nacieron con el siglo, cuando es un acumulado histórico del conocimiento humano.

2.2.3 Revisión de contenidos curriculares de matemática discreta.

Es una realidad la inclusión de la matemática discreta en los programas curriculares de ingeniería de sistemas o ciencias de la computación en todas las universidades a nivel mundial, su contenido es eminentemente matemático pero su aplicabilidad está relacionada con la algoritmia, desarrollo de software, circuitos, telecomunicaciones y criptografía entre otros.

2.2.3.1 Contexto local nacional e internacional

Se muestran tres casos. En lo local Escuela Tecnológica Instituto Técnico Central donde se realizó el trabajo de campo; en lo nacional se toma de referencia la Universidad Nacional de Colombia, reconocida por su alto nivel académico y en lo internacional el MIT Massachussets Institute of Technology que se encuentra en el ranking de las mejores universidades del mundo.

Tabla 1.

Comparativo de los Currículos de Matemática Discreta

MATERIA: MATEMÁTICA DISCRETA	**ETITC**	**UNIVERSIDAD NACIONAL DE COLOMBIA**	**MASSACHUSSETS INSTITUTE OF TECHNOLOGY**
Está incluida en el Currículo Académico de Ingeniería de Sistemas	SI	SI	SI
Código	DES-FO-05	2025963	6.04J/18.062J
Web Site	SI	NO	SI
Semestre	2do S.	4to S.	5to Semestre
Intensidad Horaria por semana	2 horas	3 horas	4 horas
Lógica Matemática	SI	SI	SI
Teoría de Conjuntos	SI	SI	SI
Relaciones	SI	SI	SI
Funciones	SI	SI	SI
Conteo	SI	SI	SI
Combinatoria	NO	SI	SI
Probabilidad	NO	NO	NO
Teoría de Grafos	SI	SI	SI
Teoría de Árboles	SI	SI	SI
Máquinas de Estado Finito	SI	SI	SI
Ruta Crítica	SI	NO	NO

Fuente: creación propia

2.2.4 Revisión libros de texto de Matemática Discreta.

En la búsqueda de libros que conforman la bibliografía de los syllabus del programa de ingeniería de sistema a nivel global y local se evidencio que los mencionados a continuación son los más estudiados, o los que componen una guía para el estudio de los temas en la asignatura matemáticas discretas.

Además, al consultar libros de matemática discreta de diversas editoriales, se han escogido tres que son los más representativos en los diferentes currículos de esta materia de los siguientes autores: (1º) Kenneth H. Rosen, (2º) Richard Johnsonbaugh y (3º) Seymour Lipschutz y Marc Lipson de la Serie SCHAUM.

Primer Libro de Matemática Discreta y sus Aplicaciones

Autor: Kenneth H. Rosen
Laboratorios AT&T
Editorial Mc Graw Hill. Quinta edición.
Año 2004 - ISBN 0-07-242434-6 - Páginas 833
Los fundamentos: Lógica y demostración, conjuntos y funciones
Algoritmos, números enteros y matrices
Razonamiento matemático, inducción y recursividad

Probabilidad discreta: Técnicas avanzadas de recuerdo, Relaciones, Grafos, Árbol, Algebra de boole, Modelos de computación

Segundo Libro de Matemática Discretas

Autor: Richard Johnsonbaugh
Editorial Prentice Hall - Sexta edición
De Paul University Chicago
Año 2005 - ISBN 970-26-0637-3 - Páginas 696

Lógica y demostraciones, El lenguaje de las matemáticas, Relaciones, Algoritmos
Introducción a la teoría de números, Métodos de conteo y el principio del palomar
Relaciones de recurrencia, Teoría de gráficas, Árboles
Modelos de redes, Algebras booleanas y circuitos combinatorios
Autómatas, gramáticas y Lenguaje, Geometría para calculo

Tercer Libro de Matemáticas Discretas

Autor: Seymour Lipschutz y Marc Lipson – Serie SCHAUM

Editorial Mc Graw Hill - Tercera edicion

ISBN 13:978-970-10-7236-3

Teoría de conjuntos, Relaciones, Funciones y algoritmos, Lógica y cálculo de preposiciones, Técnicas de conteo, Técnicas de conteo avanzadas, recurrencia, Probabilidad, Teoría de grafos, Grafos dirigidos, Árboles binarios, Propiedades de los enteros, Lenguajes, autómatas, gramáticas, Maquinas de estados finitos y máquinas de Turing, Conjuntos ordenados y retículos, Algebra booleana.

2.2.5 Desarrollo curricular de la Matemática Discreta

De acuerdo con los dos apartados anteriores de revisión curricular y revisión de libros sobre matemática discreta, existe homogenización de los contenidos y consenso en los temas tratados, por consiguiente, se infiere que hay una madurez en el tratamiento de la matemática discreta.

2.3 Didáctica de la Computabilidad

Han transcurrido ocho décadas de la formalización del modelo de computabilidad que se ha concretizado en dispositivos físicos llámense de forma genérica computadores o los dispositivos móviles denominados celulares; los cuales son simplemente formas que toma “la máquina de

Turing", ubicando una cifra conservadora en 5.000 millones de unidades en el mundo, ello muestra que es un modelo muy exitoso.

Con esta revolución computacional, sería la segunda vez que el trasfondo científico ha sido matemático; la diferencia con la primera vez, cuando el cálculo infinitesimal (matemática continua) fue el protagonista de la revolución industrial, mientras hoy es la matemática discreta, el fundamento de los desarrollos computacionales del siglo XXI.

Una generación que vive inmersa en el modelo computacional tiene la obligación de adentrarse en la trastienda de ese mundo; no es fácil por el alto nivel de complejidad y matematización que se requiere, pero ese es el reto; se necesita una didáctica de la computabilidad en general y de la matemática discreta en particular.

La computabilidad es un concepto muy propio del siglo XXI, debido al gran número de "máquinas de Turing" en la vida cotidiana; así se va construyendo una manera de pensar y un modo de actuar computablemente, pensar que todo es solo oprimir un botón o explicar modelos solo con inputs y outputs.

Las máquinas de Turing son parte del paisaje tecnológico, hacen parecer natural e intuitivo el concepto de computabilidad, pero desafortunadamente el éxito del modelo, no deja ver todo el bagaje científico – técnico que lleva incorporado como resultado de siglos y siglos de conocimiento acumulado.

Dentro de este contexto, la enseñanza de la computabilidad tiene dos aspectos: el formativo y el instrumental; por lo tanto, la didáctica cumple un papel importante en el proceso de apropiación del concepto de computabilidad, en la búsqueda de facilitar el aprendizaje de dichos conceptos. Estudiar la computabilidad en función de los principios fundamentales que permiten y posibilitan la era computacional.

Según Guy Brousseau (1998), la didáctica de las matemáticas es la responsable del estudio científico (teórico, empírico y experimental) de las condiciones en las cuales una población estudiantil se culturiza con cualquier tema de las matemáticas; proporcionando los instrumentos teóricos de sus proyectos y trata de dar cuenta de sus resultados. Es con la matemática discreta que se han hecho los más importantes aportes a la era de la computación, hay una especificidad en ese camino construido que es muy diferente al desarrollo de la matemática clásica, la matemática continua y es el concepto de computabilidad el que engloba las búsquedas y los logros.

Las matemáticas en general tienen un papel muy relevante en lo formativo y la matemática discreta de forma específica debe estudiar las estructuras matemáticas que es un tema muy abstracto; sin embargo, es aquí donde surge la problemática cuando se trata de una enseñanza tecnológica; es reconocer que no se está enseñando a matemáticos, se imparte una formación a ingenieros, donde se requiere una aplicabilidad de dicho conocimiento.

Cada uno de los temas de matemática discreta ha tenido una madurez en su formalismo, se debe dimensionar lo que se presenta en cada tema, contextualizándolo a la ingeniería de sistemas.

La siguiente tabla describe los componentes del concepto de computabilidad y la evidencia de aprendizaje que se espera que el estudiante de ingeniería de sistemas desarrolle frente a estos.

Tabla 2.

Componentes del concepto de computabilidad

Componente/dimensión	**Evidencia de aprendizaje**
Máquina de Turing	-Establece el concepto matemático de una máquina de Turing. -Identifica las componentes de una máquina de Turing. -Reconoce el funcionamiento de una máquina de Turing.

Funciones Recursivas	-Reconoce el concepto de función y sus propiedades. -Reconoce el concepto de relación y sus características. -Identifica los tipos de funciones en inyectiva, sobreyectiva y biyectiva. -Establece las propiedades de función computable. -Identifica el concepto de función recursiva.
Lógica Computable	-Identifica el concepto de computabilidad asociado a la lógica. -Asocia el modelo de circuitos con lógica. -Reconoce las operaciones entre conjuntos.
Algoritmia	-Identifica las características de un algoritmo. -Comprende las características de un diagrama de flujo. -Construye de manera experimental un algoritmo lógico.

Fuente: creación propia

Capítulo 3. Metodología

En este capítulo se muestra metodológicamente la naturaleza mixta de la investigación, se explica el alcance explicativo y su diseño secuencial; se describen la población, la muestra, los instrumentos de recolección de datos y el método de análisis.

3.1 Enfoque de la Investigación.

El estudio se inscribe metodológicamente bajo las premisas de las investigaciones mixtas. Se busca comprender un fenómeno teniendo en cuenta su complejidad, desde una mirada compartida y complementaria, abordando diferentes niveles del problema. En este caso se pretende describir las dificultades asociadas a la apropiación del concepto de "computabilidad" de los estudiantes del curso de «matemáticas discretas», a partir de una aproximación "objetiva", guiada por el método cuantitativo; y una subjetiva, guiada por el método cualitativo; para finalmente hacer una correlación entre los resultados cuantitativos y cualitativos, buscando generar una comprensión holística del fenómeno estudiado.

De acuerdo con Todd, Nerlich y McKeown (2004), "las investigaciones mixtas producen datos más "ricos" y variados, de diversas fuentes" (p.45), que permiten, según Hernández y Mendoza (2018), "una "riqueza interpretativa" o un "mayor poder de entendimiento". (p. 550).

De otro lado, y para el caso de esta investigación, Hernández, Fernández, y Baptista, (2014) expresa: "la aplicación de los métodos cualitativo y cuantitativo permite complementación, en la medida que se busca que los resultados de un método clarifiquen el entendimiento de los resultados del otro método". (p. 551).

La primera fase se realiza mediante la aplicación de una prueba al 100% de la población (26 estudiantes) para así obtener unos datos que permiten una clasificación en niveles (bajo, medio y alto) de apropiación al concepto de computabilidad, además por cada dimensión mencionada en el capítulo anterior se obtienen datos puntuales a analizar. Respecto a la segunda fase se realizó una entrevista a un grupo focal de seis estudiantes, con preguntas desde su experiencia personal hacia el proceso de aprendizaje incluyendo particularidades como: diferenciación entre conceptos, la manera en la cual proyecta los procedimientos y las dificultades asociadas a los mismos entre otras. Por último, en la tercera fase se correlacionan los resultados logrados por el método cuantitativo con los obtenidos por el método cualitativo, para tener una mejor comprensión e integración del análisis.

3.2 Alcance de la Investigación

Teniendo en cuenta lo anterior, el alcance de la investigación es explicativo, en la medida que se intenta especificar las características del proceso de apropiación del concepto de “computabilidad”, visibilizando su relación con los factores que, según los participantes, inciden en el proceso. Dicho de otro modo y en el marco del planteamiento del problema, se pretende dar respuesta a las siguientes preguntas: ¿cuál es el nivel de apropiación del concepto de “computabilidad” en los estudiantes del curso “matemáticas discretas” del programa de Ingeniería de Sistemas? ¿cuáles son los factores que, según los estudiantes, afectan su aprendizaje del concepto de “computabilidad”?

3.3 Diseño de la Investigación.

El diseño se realizó de forma secuencial, para alcanzar los logros de los objetivos que se requieren de acuerdo con la investigación.

> Al plantear unos lineamientos que enmarcan el diseño de la investigación se establece lo siguiente: “el diseño secuencial se caracteriza por una primera etapa donde se recolectan y analizan datos cuantitativos y en una segunda fase se recaban y analizan datos por el otro método” (Hernández, Fernández, & Baptiste, 2014, p.559).

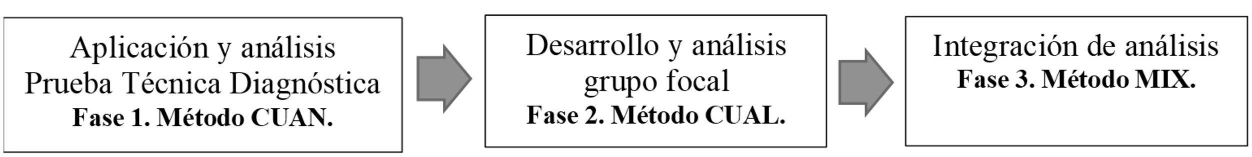
Por lo tanto, el diseño secuencial aplicado a la investigación se realiza en tres momentos. El primero, una aproximación “objetiva” guiada por el método cuantitativo; el segundo, una aproximación “subjetiva”, guiada por el método cualitativo, y en el tercero se correlacionaron los resultados cuantitativos y cualitativos, buscando generar una comprensión holística del fenómeno estudiado.

Se aplicó una prueba en la primera fase con respecto a los siguientes temas de: máquina de Turing, funciones recursivas, algoritmia y lógica computable. Posteriormente a partir de los resultados de esta prueba, en la segunda fase se diseña el protocolo de desarrollo para la aplicación de la entrevista focal. El resultado de la prueba del primer momento determina el criterio de selección de los participantes que harán parte de la segunda prueba.

> Teniendo en cuenta la naturaleza mixta de la investigación y los tiempos del estudio, se optó por una ejecución secuencial, que implicó, de acuerdo con recolectar y analizar, en una primera fase, los datos cuantitativos, con el fin de orientar, la recolección y análisis de los datos cualitativos, en una segunda fase. Por último, se integraron los análisis de las dos fases (tercera fase), (Hernández e & Mendoza 2018, p.556).

Tal como se muestra en la figura 11.

Figura 11. *Fases de la metodología*



Fuente: Construcción de la autora

3.4 Población

La Escuela Tecnológica Instituto Técnico Central (ETITC) fundada en 1905 es una institución de carácter público de educación superior ubicada en la ciudad de Bogotá. La cual ofrece seis pregrados en ingeniería entre ellas, Ingeniería de Sistemas la cual está constituida por ciclos. Primer ciclo, Técnica profesional en computación, segundo ciclo Tecnología en desarrollo de software y el tercer ciclo Ingeniería de sistemas.

La población en estudio corresponde a 26 estudiantes, distribuidos por género en 8 mujeres y 18 hombres, cuyas edades se encuentran en un rango de 18 - 21 años; todos ellos son estudiantes de la asignatura Matemáticas discretas la cual se cursa en segundo semestre de la carrera ingeniería de sistemas, pero al ser por ciclos la ingeniería, el segundo semestre corresponde particularmente al ciclo técnica profesional en computación. La prueba de la fase uno y dos se aplicó en la semana dieciséis de las dieciocho semanas que conforman el semestre.

Muestra 1

La muestra 1 está conformada por el 100% de la población, que corresponde a los 26 estudiantes de la asignatura "Matemáticas discretas". Se busca, siguiendo los parámetros del método cuantitativo, generalizar los resultados con respecto a un fenómeno (niveles de apropiación del concepto de computabilidad), a través de la aplicación de una prueba de conocimientos.

Se hizo el estudio con el 100% de la población, determinando así una mejor aproximación en el estudio, que según Hernández y Mendoza (2018) plantean que al "abarcar la totalidad de la población los resultados son de gran validez y confiabilidad" (p.219)

Muestra 2

Es una muestra direccionada eligiendo unos criterios para su selección. Sobre esta población se realiza el estudio cualitativo de la investigación.

En esta segunda fase la muestra está conformada por seis estudiantes del total de la población, quienes se denominaron: "grupo focal". Estos participantes se seleccionaron siguiendo los siguientes criterios:

1. Sexo: representatividad de ambos sexos en la recolección de datos.
2. Niveles de apropiación: la representatividad por cada uno de los niveles de apropiación establecidos: bajo – medio – alto.

Tabla 3.

Descripción de la muestra

Muestra 1 Aplicación prueba de conocimiento (26 estudiantes)	100 % de los estudiantes de la asignatura “Matemáticas discretas”				
Muestra 2 **Participación grupo focal (6 estudiantes)**	**Criterios de selección**				
	Sexo		**Nivel de apropiación**		
Participantes	**M**	**F**	**Bajo**	**Medio**	**Alto**
Participante 1	x		x		
Participante 2		x	x		
Participante 3	x			x	
Participante 4		x		x	
Participante 5	x				x
Participante 6		x			x

Fuente: creación propia

En la tabla 3 se observa cómo se conformó la muestra. Muestra 1 con el 100% de los estudiantes de la asignatura “Matemáticas Discretas” y la muestra 2, con seis participantes que cumplan los requisitos de sexo es decir 3 hombres y tres mujeres de los cuales dos están en el nivel alto, dos en el nivel medio y dos en el nivel bajo de la prueba de conocimiento realizada en la fase uno.

3.5 Variables

Dando alcance a los objetivos propuestos en la investigación y de acuerdo con el desarrollo del marco teórico se identificaron para la computabilidad cuatro variables estructurales y 14 subvariables que se asocian directamente con los objetivos de la investigación, estas fueron el insumo para la construcción del instrumento del cuestionario y la entrevista.

Tabla 4.

Cuestionario

Objetivos de investigación	Categoría	Sub categoría	Instrumento
Determinar los niveles de apropiación del concepto de "computabilidad" en los estudiantes del curso "matemáticas discretas" del programa de Ingeniería de Sistemas.	Máquina de Turing	Concepto matemático de una máquina de Turing.	Cuestionario
		Componentes de una máquina de Turing.	
		Funcionamiento de una máquina de Turing.	
	Funciones Recursivas	Concepto de relación	
		Concepto de función	
		Tipo de funciones	
		Función computable	
		Concepto de función recursiva	
	Lógica computable	Lógica	
		Circuitos lógicos	
		Conjuntos	
	Algoritmia	Características de un algoritmo	
		Características diagrama de flujo	
		Construcción de un algoritmo	

Fuente: creación propia

Tabla 5.

Entrevista grupal

Objetivos de investigación	Categoría	Sub categoría	Instrumento
Describir las dificultades asociadas a la apropiación del concepto de "computabilidad de los estudiantes del curso "Matemáticas discretas" del programa Ingeniería de Sistemas	Máquina de Turing	Concepto matemático de una máquina de Turing.	
		Componentes de una máquina de Turing.	
		Funcionamiento de una máquina de Turing.	
	Funciones Recursivas	Concepto de relación	
		Concepto de función	
		Tipo de funciones	
		Función computable	
		Concepto de función recursiva	
		Lógica	

	Lógica computable	Circuitos lógicos	
		Conjuntos	Entrevista grupal
	Algoritmia	Características de un algoritmo	
		Características diagrama de flujo	
		Construcción de un algoritmo	

Fuente: creación propia

3.6 Instrumentos

El primer instrumento es una prueba de conocimientos (Apéndice C) para valorar el nivel de apropiación de los estudiantes con respecto al concepto de computabilidad. La prueba tiene 38 ítems agrupados en cuatro dimensiones diferentes (Máquina de Turing, funciones recursivas, lógica computable y algoritmia), cada una de las dimensiones constituye un subconjunto del tema de computabilidad tal como se explicó en el capítulo dos. Y cada dimensión tiene asociado unas evidencias de aprendizaje desde el conocimiento.

La nomenclatura implementada en el test se presenta en la tabla 6.

Tabla 6.

Nomenclatura

Variable	**Subvariable**	**Pregunta asociada**
V.1	**V.1.2**	**V.1.2.3**
Cada variable esta numerada solo con un índice. Para el caso particular hace referencia a la variable uno.	El primer índice de subvariable hace referencia a que variable corresponde y el segundo índice al número de subvariable. Para el caso particular pertenece a la variable uno y es la segunda subvariable	Las preguntas tienen tres subíndices, el cual el primer índice corresponde a la variable asociada, el segundo subíndice a la subvariable asociada y el tercero es el índice del número de pregunta. Para el caso particular pertenece a la variable uno, subvariable dos y es la tercera pregunta.

Fuente: creación propia

A continuación, se presentan cuatro tablas que desglosan la ponderación a cada variable y el peso asignado a cada ítem.

En la tabla 7 se presenta la variable máquina de Turing se divide el 100% del 25% entre las preguntas de la variable, obteniéndose un peso del 10% para cada ítem. Es decir, si el estudiante responde las 10 preguntas de manera correcta obtendrá el 25% del total de la prueba.

Tabla 7.

Primera variable

Variable	Peso	Subvariable	Peso	Pregunta	Peso
Máquina de Turing	25%	Concepto matemático de una máquina de Turing.	33.3%	V.1.1.1	10%
				V.1.1.2	10%
				V.1.1.3	10%
		Componentes de una máquina de Turing.	33.3%	V.1.2.1	10%
				V.1.2.2	10%
				V.1.2.3	10%
		Funcionamiento de una máquina de Turing	33.3%	V.1.3.1	10%
				V.1.3.2	10%
				V.1.3.3	10%
				V.1.3.4	10%
TOTAL			**100%**	**10 preguntas**	**100%**

Fuente: creación propia

Para la segunda variable funciones recursivas, presentada en la tabla 8, se divide el 100% del 25% entre las preguntas de la variable, obteniéndose un peso del 9,1% para cada ítem. Es decir, si el estudiante responde las 11 preguntas de manera correcta obtendrá el 25% del total de la prueba.

Tabla 8.

Segunda variable

Variable	Peso	Subvariable	Peso	Pregunta	Peso %
Funciones Recursivas	25%	Concepto de relación		V.2.1.1	9.1
				V.2.1.2	9.1
		Concepto de función		V.2.2.1	9.1
				V.2.2.2	9.1
		Tipo de funciones		V.2.3.1	9.1
				V.2.3.2	9.1
				V.2.3.3	9.1

		Función computable		V.2.4.1	9.1
				V.2.4.2	9.1
		Concepto de función recursiva		V.2.5.1	9.1
				V.2.5.2	9.1
TOTAL			**100%**	**11 preguntas**	**100%**

Fuente: creación propia

La tabla 9 presenta la tercera variable lógica computable donde se divide el 100% del 25% entre el total de las preguntas, obteniendo un peso del 11,11% para cada ítem. Es decir, si el estudiante responde las 9 preguntas de manera correcta obtendrá el 25% del total de la prueba.

Tabla 9.

Tercera variable

Variable	Peso	Subvariable	Peso	Pregunta	Peso %
		Lógica		V.3.1.1	11,11
				V.3.1.2	11,11
				V.3.1.3	11,11
				V.3.1.4	11,11
Lógica	25%	Circuitos lógicos		V.3.2.1	11,11
computable				V.3.2.2	11,11
		Conjuntos		V.3.3.1	11,11
				V.3.3.2	11,11
				V.3.3.3	11,11
TOTAL			**100%**	**9 preguntas**	**100%**

Fuente: creación propia

La cuarta variable presentada en la tabla 10, denominada algoritmia se divide el 100% del 25% entre el total de las preguntas, obteniendo un peso del 12,5% para cada ítem. Es decir, si el estudiante responde las 8 preguntas de manera correcta obtendrá el 25% del total de la prueba.

Tabla 10.

Cuarta variable

Variable	Peso	Subvariable	Peso	Pregunta	Peso %
				V.4.1.1	12,5
		Características de un		V.4.1.2	12,5
Algoritmia		algoritmo		V.4.1.3	12,5
	25%	Características		V.4.2.1	12,5
		diagrama de flujo		V.4.2.2	12,5
				V.4.2.3	12,5
		Construcción de un		V.4.3.1	12,5
		algoritmo		V.4.3.2	12,5
TOTAL			**100%**	**8 preguntas**	**100%**

Fuente: creación propia

La tabla 11 establece la relación entre puntuación obtenida en la prueba de conocimientos y el nivel de apropiación del concepto de computabilidad.

Tabla 11.

Rango de puntuación

Nivel	Rango %
Bajo	0 – 30
Medio	31 – 70
Alto	71 – 100

Fuente: creación propia

Cuestionario (grupo focal)

Se realiza un cuestionario (Apéndice C) a través de un conjunto de preguntas diseñadas a partir de las mismas variables y subvariables con el fin de lograr una mayor comprensión desde la experiencia y significado que le da el estudiante al proceso.

El instrumento dos (Apéndice C) es una entrevista que se realiza únicamente al grupo focal comprendido por 6 estudiantes. Los cuales son 2 de nivel bajo, 2 de nivel medio y dos de nivel alto Millan y Sally (2005) “En la investigación mediante test, el investigador selecciona una muestra de la población les administra un cuestionario o realiza entrevistas para recoger información sobre las variables de interés” (p.292).

3.7 Procedimientos

En esta sección se describen los procedimientos, las técnicas e instrumentos utilizados, este proceso se realizó en cinco etapas para obtener, analizar e interpretar los datos.

Inicialmente se definió una matriz que permitió pasar de la categoría al diseño del ítem de la pregunta ver Apéndice B, una vez diseñado el instrumento se realizó la validación por parte de dos expertos ver Apéndice D, después de la validación se procedió a la realización de un pilotaje con 10 estudiantes de la Corporación Universitaria Republicana, los cuales cumplen las mismas características de los estudiantes de la ETITC en tanto que son estudiantes de ingeniería de sistemas de segundo semestre y además cursando la misma asignatura matemáticas discretas.

Luego del pilotaje, usando Google Forms se envió el cuestionario vía correo electrónico a cada uno de los 26 estudiantes cuya primera pregunta tiene las opciones de SI y de No; en caso del SI, es porque el estudiante acepta el “consentimiento informado”, en caso de responder NO, el estudiante termina automáticamente el proceso y no podrá contestar las preguntas del cuestionario, ver Apéndice A.

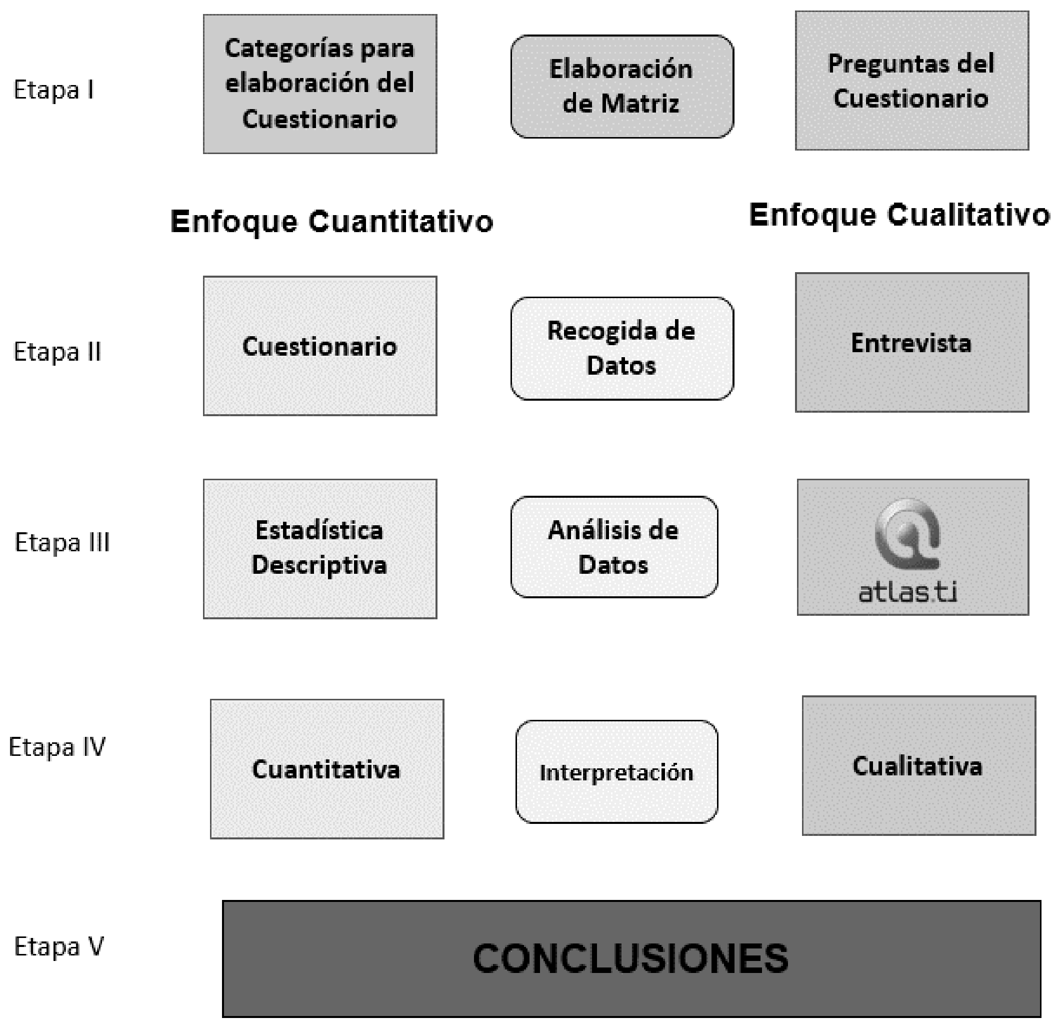
El tiempo tomado para las respuestas de los estudiantes fue de dos semanas. Con el recibo de las respuestas de los cuestionarios debidamente validados se da fin a la etapa del proceso cuantitativo.

A continuación, se comienza la etapa cualitativa citando en total a seis estudiantes del grupo focal, dos de alto nivel, dos de nivel medio y dos de nivel bajo a una sesión en línea mediante la herramienta TEAMS en donde se desarrolló la entrevista siguiendo el protocolo del grupo focal ver Apéndice A.

Una vez se validó que se tienen las grabaciones en formato digital, se pone fin de la primera etapa respecto al trabajo cualitativo.

El modelo general de este proceso se presenta en las siguientes cinco etapas.

Figura 12. Etapas del proceso de la investigación.



Fuente: construcción de la autora.

3.7.2 Análisis de datos

Una vez recibido el 100% de los cuestionarios, se procedió a realizar la calificación de cada uno, y se estructuraron los tres niveles: alto, medio y bajo. De cada uno de los anteriores grupos se escogieron 6 estudiantes: dos de nivel alto, dos de nivel medio y dos de nivel bajo; tres mujeres y tres hombres. Se citaron a una entrevista grupal respondiendo a 14 preguntas, con un tiempo de duración de 3 horas, la entrevista se transcribió y una vez transcrita, se subió al programa Atlas.ti.

Los datos recolectados se analizaron mediante el procedimiento de codificación abierta a través del software Atlas.ti.

En un primer nivel, se asignaron códigos (etiquetas) a los textos transcritos, descompuestos en citas o fragmentos de texto más cortos. En un segundo nivel se agruparon dichos códigos en función de las variables y subvariables definidas previamente. Por último, se propuso un esquema interpretativo por categoría que tuvo en cuenta la recurrencia y relación entre códigos, con el fin de atender a los objetivos propuestos y tratar de dar respuesta a las preguntas de investigación. En resumen y como lo afirman Hernández et al. (2010) “se dotó de estructura a los datos no estructurados, resultantes de la aplicación del instrumento.” (p.574).

Capítulo 4. Resultados y Análisis

En este capítulo se presentan los niveles de apropiación del concepto de computabilidad en estudiantes de ingeniería de sistemas, así como las dificultades que experimentan en el proceso de enseñanza - aprendizaje. Atendiendo al primer objetivo específico, se expone el análisis de los resultados de la prueba de conocimiento, con el fin de identificar los niveles de apropiación del concepto de "computabilidad" en los estudiantes del curso "matemáticas discretas" del programa de Ingeniería de Sistemas (mirada cuantitativa). Posteriormente, y alineado al segundo objetivo específico, se plantea una comprensión de las dificultades asociadas a la apropiación del concepto de "computabilidad" a partir del reconocimiento de la voz de los estudiantes (mirada cualitativa). Por último, y en el marco del objetivo general de la investigación, se relacionan las dificultades de los estudiantes con los niveles de apropiación identificados (mirada mixta).

4.1 Niveles de apropiación del concepto de computabilidad

La "computabilidad" no es un concepto acabado, aún en el siglo XXI se encuentra en completo desarrollo. Para efectos de este trabajo y en consideración de la formación disciplinar de la ingeniería de sistemas y desde la particularidad de la matemática discreta se hace una aproximación al concepto de computabilidad desde los siguientes cuatro enfoques: máquina de Turing, funciones recursivas, lógica computable y algoritmia.

Esta manera de hacer la aproximación genera cuatro variables que son representativas en conjunto porque estructuran una forma de hacer la apropiación del concepto de "computabilidad"; aunque no agotan el concepto y existen otras muchas formas de hacer dicha aproximación.

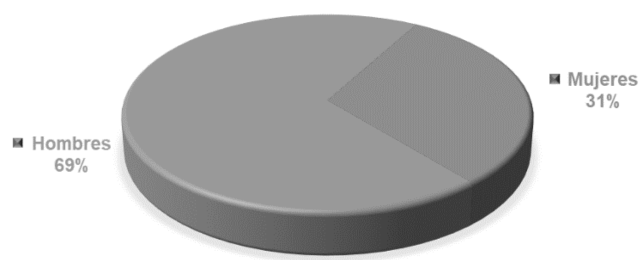
Los resultados con respecto al nivel de apropiación del concepto de "computabilidad", se obtuvieron aplicando una prueba de conocimientos estructurada sobre las cuatro variables mencionadas, teniendo en cuenta la siguiente equivalencia conceptual: máquina de Turing =

mecanismo; funciones recursivas = formalismo matemático; lógica computable = inferencia; y algoritmia = método, alcances y limitaciones de la computabilidad.

4.1.1 Caracterización de los participantes

Los participantes que respondieron la prueba de conocimiento son estudiantes de la Escuela Tecnológica Instituto Técnico Central (ETITC), quienes cursan segundo semestre del programa ingeniería de sistemas. En total, fueron 26 alumnos distribuidos en 8 mujeres (31%) y 18 hombres (69%), como se observa en la figura 13. El perfil socioeconómico de la institución está conformado principalmente por jóvenes entre los 18 y 21 años; de estrato 1, 2 y 3 de la ciudad de Bogotá.

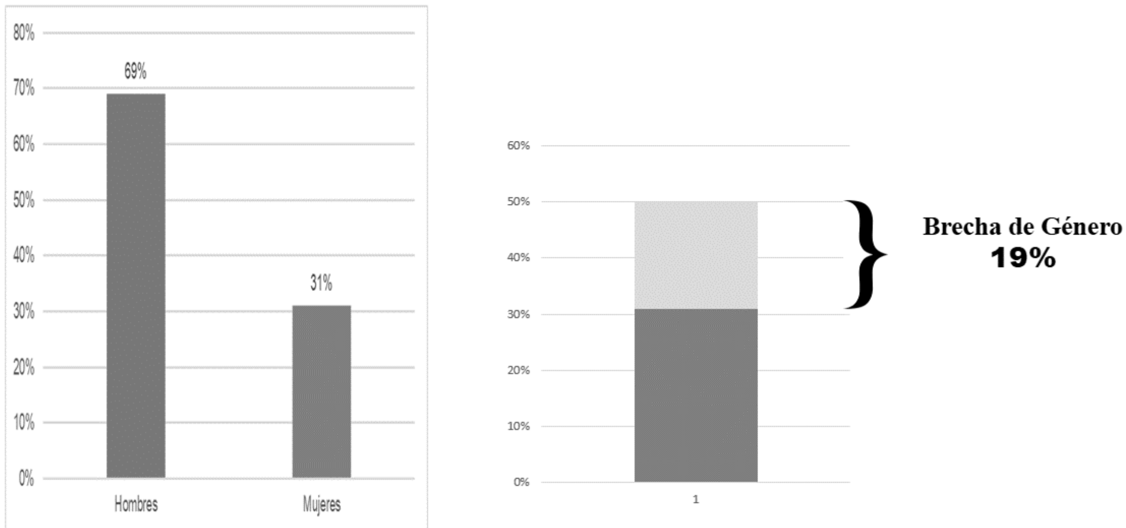
Figura 13. Distribución por género



Fuente: construcción de la autora

La participación femenina, en términos porcentuales, demuestra que existe una brecha de género del orden del 19%, tal como se muestra en la figura 14; siendo esta la diferencia entre el 50% que debiera ser la población femenina y el 31% que es el existente. Lo anterior, según Patiño (2020), "muestra las diferencias por género en la educación superior pública colombiana que corresponde a un 31,64%". La población del presente trabajo mantiene dicha similitud al porcentaje presentado en el estudio realizado a nivel nacional entre los años 2001 al 2018 titulado: "Una maratón en tacones: la brecha de género en ciencia y tecnología en la educación superior colombiana." (p.10). Es importante no naturalizar dicha brecha en el sector de ciencia y tecnología, más cuando dicha desigualdad ha logrado superarse en cuanto al acceso a la educación superior en muchas otras áreas.

Figura 14. Brecha de participación por género con la población seleccionada



Fuente: Construcción de la autora.

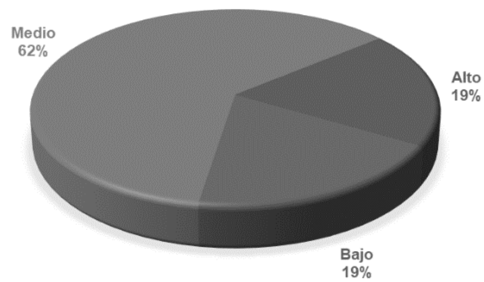
4.1.2 Niveles de apropiación

La figura 15 presenta los porcentajes globales obtenidos en la prueba de conocimientos, la cual constó de 38 preguntas realizadas a los 26 participantes. Se definió una escala de tres niveles: bajo, medio y alto. El nivel bajo se estableció entre 0 y el 57% de asertividad, el nivel medio entre el 58% y el 79% de asertividad, el nivel alto entre el 80% y el 100% de asertividad.

Los resultados globales muestran un 19% en el rango bajo, 62% se encuentra en el rango medio y 19% en el rango alto.

Si sumamos el 62% del rango medio con el 19% del rango alto, obtenemos un 81% de la población con un nivel de apropiación adecuado en el tema de "computabilidad", estos resultados son acordes con lo que se espera para estudiantes de segundo semestre que según lo establecido en el syllabus de la institución (2020), "sus competencias deben estar fundamentadas en el hacer a partir de las destrezas y habilidades, aplicando su formación integral en las ciencias básicas de la ingeniería” (p.1) y más relevante en consideración a que son temáticas transversales en la ingeniería de sistemas.

Figura 15. Escala de distribución global por niveles de apropiación

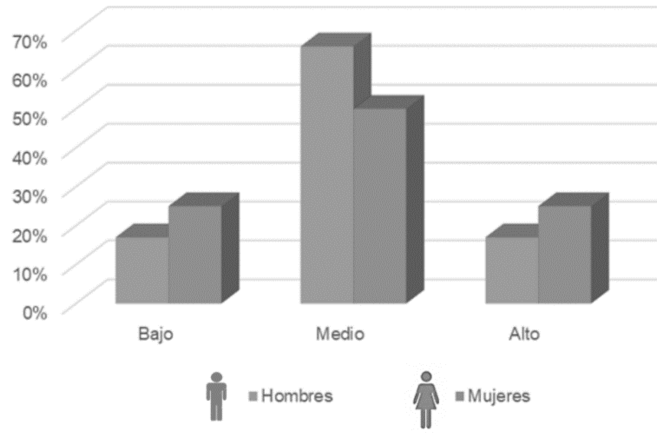


Fuente: construcción de la autora

El comparativo por género se muestra en la figura 16; en el nivel alto se evidencia que a las mujeres tienen un porcentaje de apropiación del 25%, frente al 16,6% en los hombres; esta diferencia de 8 puntos es bastante significativa más cuando el área de ingeniería tradicionalmente es concebida como “propia” del género masculino. Lo anterior contrasta con el pensamiento que generalmente se tiene sobre el rendimiento de las mujeres en este campo. Como bien lo mencionan Aguiar, Gutiérrez, Lara y Villalpando (2011) “Esto sugiere un mejor desempeño promedio de las mujeres tanto en su calidad de estudiantes como de docentes en el campo de la ingeniería de sistemas” (p.14)

En el nivel medio; corresponde el 66,6% para hombres y para mujeres el 50%. Con ello se infiere que los hombres tienen su mayor población en el nivel medio, superando a las mujeres en 16 puntos porcentuales.

Figura 16. Comparación por género - distribución global por niveles de apropiación



Fuente: construcción de la autora.

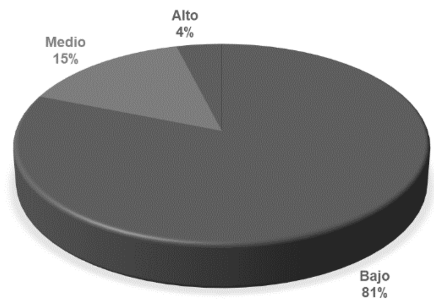
4.1.2.1 Máquina de Turing

La variable "máquina de Turing" presenta los niveles más bajos de apropiación del concepto de "computabilidad" con respecto a las otras tres variables (funciones recursivas, lógica computable y algoritmia). El 81% de los participantes se encuentran el rango "bajo", tal como se muestra en la figura 17.

Se pretende hacer una aproximación al pensamiento de Turing desde el texto mismo del paper: "On computable numbers with an application to the entscheidungsproblem", escrito en el año 1936, donde quedó plasmado su conceptualización de computabilidad, recuperando el sentido original de su creación. La dificultad y complejidad del tema es reconocido por varios autores, como lo menciona Castro (2011), "evidencia la problemática en la comprensión de los procesos lógico – deductivos de la computabilidad; de ahí la importancia de la didáctica para poder superar estos limitantes." (p.4)

Como lo expresan Rosenfeld y Irazábal (2013), "se utilizan las máquinas de Turing como modelo de ejecución de los algoritmos, el tiempo como recurso computacional para medir la complejidad de un problema, y la semántica operacional como técnica para especificar formalmente un lenguaje de programación.” (p. 8).

Figura 17. Niveles de apropiación de la variable máquina de Turing



Fuente: construcción de la autora.

4.1.2.2 Funciones recursivas

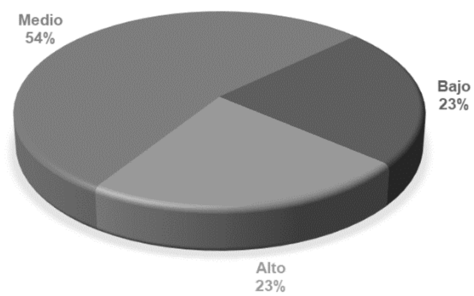
La figura 18 presenta los resultados de la variable “funciones recursivas”, donde el 54% de la población se encuentra en el rango medio. Además, presenta una mejora con respecto a la variable “Máquina de Turing”, donde el rango de estudiantes en el nivel bajo se reduce del 81% al 23%.

La figura 18 muestra un resultado simétrico donde el 23% de la población tuvo un posicionamiento en el rango bajo, 54% en el nivel medio y 23% en el nivel alto.

La recursividad es un pilar de la computabilidad, de las cuatro variables es la que requiere un análisis, comprensión y aplicabilidad del más alto nivel de formalización matemática. Los niveles encontrados pueden explicarse por los conocimientos de la teoría de funciones adquiridos por los estudiantes en su educación media, de acuerdo con Trejos (2017) “Se encontró que los alumnos confieren suprema importancia a la relación entre el nuevo conocimiento que se les explica y sus nexos con los conocimientos ya adquiridos” (p.63). Lo nuevo en esta etapa de formación es la diferenciación que existe entre “funciones recursivas” y “computabilidad”, como se mencionó en el capítulo del marco teórico.

> La computabilidad de Turing se complementa con la de Alonzo Church en cuanto a funciones recursivas y es importante diferenciar estos dos conceptos porque en los años 30 se trató de asociar el concepto de computabilidad con el de recursividad y para fines del siglo XX estos conceptos tenían cada uno vida propia y no tienen igual significado. (Mota, 2015, p,159).

Figura 18. Niveles de apropiación de la variable funciones recursivas

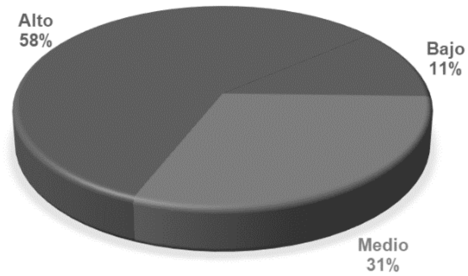


Fuente: construcción de la autora

4.1.2.3 Lógica computable

En la variable “lógica computable” hay una mejora sustantiva en el nivel de apropiación; como se muestra en la figura 19, el 58% de la población tiene un nivel alto, evidenciando una mejor apropiación de los temas como lógica proposicional, álgebra booleana, operaciones entre conjuntos, circuitos lógicos y el concepto de computabilidad asociado a estos temas. La lógica moderna que se utiliza en el modelo computacional existente forma parte del primer tercio del programa curricular (2020) de la materia “matemáticas discretas”.

Figura 19. Niveles de apropiación de la variable lógica computable



Fuente: construcción de la autora.

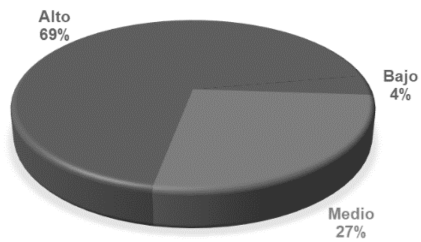
4.1.2.4 Algoritmia

La variable "algoritmia" presentó el más alto porcentaje de apropiación de las cuatro variables, obteniendo un 69% de asertividad como se muestra en la figura 20; este tema corresponde al desarrollo de la metodología para resolver un problema; son pasos lógicos y finitos que se deben construir para encontrar la solución de un problema. De acuerdo con Barchini (2008), "Turing construye mediante algoritmia su concepto de computabilidad" (p.11).

Este tema debe ser asumido no solamente desde la definición clásica de algoritmo; es necesario plantear a los alumnos la definición de algoritmos desde otros formalismos como el expuesto desde la teoría conjuntista.

> La siguiente cuádrupla expresa la definición de algoritmo desde la teoría de conjuntos: $\boldsymbol{A}= <\boldsymbol{Q}, \boldsymbol{E}, \boldsymbol{S}, \boldsymbol{F}>$, donde ***Q*** es el conjunto de todos los elementos simples y K-fórmulas que pueden describir el cálculo, ***E*** es un subconjunto de ***Q*** que hace referencia a los datos de entrada, ***S*** también es un subconjunto de ***Q*** cuyos elementos son los resultados y ***F*** se refiere a la regla del cálculo. Esta regla, a partir de un elemento q_0, genera una sucesión $q_0, q_1, q_2, q_3, \ldots$ tal que: $q_{n+1},= F(q_n)$ donde n € N, q_0 € $E \subset Q$. (Fernández y Sáez, 1987, p. 313).

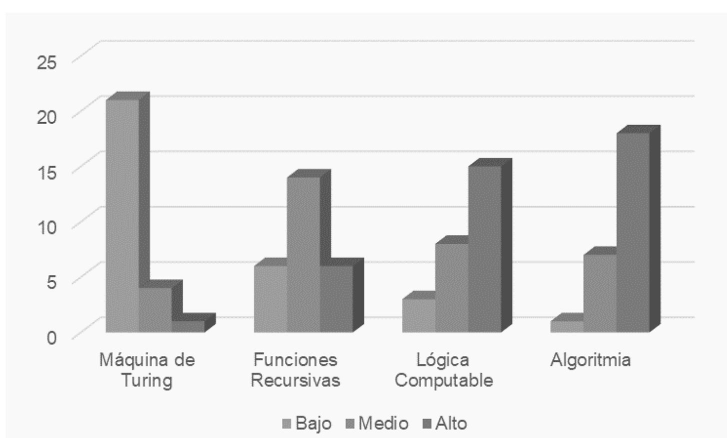
Figura 20. Niveles de apropiación de la variable algoritmia



Fuente: construcción de la autora.

Por último, se hace una comparación general de las cuatro variables como se observa en la figura 21; siendo importante aclarar que estas temáticas son muy diferentes en cuanto a definición, comprensión y aplicación. Se integran en la "matemática discreta" y es uno de los fundamentos en la formación integral del futuro ingeniero de sistemas. En esta comparación general se aprecia lo disímil del comportamiento en las cuatro variables que forman una unidad en cuanto a computabilidad se refiere.

Figura 21. Comparativo global de los cuatro niveles de apropiación



Fuente: construcción de la autora.

El análisis del grupo focal permite comprender mejor los resultados anteriormente descritos, por qué, en la variable “Máquina de Turing”, ¿los estudiantes tuvieron el menor porcentaje de apropiación? ¿por qué, en la variable “algoritmia”, los estudiantes tuvieron el mejor porcentaje de apropiación? Estos interrogantes surgen desde la reflexión de la práctica pedagógica asociada con los tiempos de dedicación que cada una de las temáticas (variables) tuvo durante el desarrollo del curso “matemáticas discretas”.

4.2 Dificultades a partir del reconocimiento de la voz de los estudiantes

El análisis cualitativo del presente estudio se realizó mediante los datos obtenidos a partir de una entrevista grupal de cinco estudiantes; la cual tuvo 15 preguntas orientadoras alrededor de las 4 variables que estructuran el concepto de “computabilidad”.

Para cada una de las cuatro variables: “máquina de Turing”, “funciones recursivas”, “lógica computable” y “algoritmia”, se identificaron las dificultades en la apropiaciòn del concepto de computabilidad asociado al segundo objetivo especifico de la investigación. El procesamiento de la información se realizó mediante el software Atlas.ti

4.2.1 Caracterización de los estudiantes del grupo focal

El grupo de “matemática discreta” estaba conformado por 26 estudiantes (8 mujeres y 16 hombres), son alumnos de la Escuela Tecnológica Instituto Técnico Central - ETITC, quienes al momento de la entrevista cursaban segundo semestre del programa ingeniería de sistemas.

Para la muestra 2, el siguiente fue el criterio de selección para escoger el grupo focal de 5 estudiantes, tres hombres y dos mujeres; primeramente, se tuvo en cuenta que ya se había presentado la prueba de conocimientos y de acuerdo con el puntaje que obtuvo cada estudiante el grupo de 26 alumnos se había dividido en alto, medio y bajo. El proceso se realizó aleatoriamente de la siguiente manera: del grupo alto se escogieron dos alumnos (un hombre y una mujer), del grupo medio igualmente dos alumnos (un hombre y una mujer) y del grupo bajo se escogió un hombre; y así de esta manera quedó conformado el grupo focal con cinco estudiantes. La entrevista focal se realizó el 18 de mayo del año 2020 y el tiempo de duración fue de 49 minutos. (Ápendice E)

4.2.1.1 Máquina de Turing

La máquina de Turing es el modelo computacional más elaborado y complejo hasta la fecha, y este no ha sido superado a pesar de los 84 años transcurridos desde su creación. Con respecto a la variable "Máquina de Turing", se indagó sobre las subvariables: "concepto matemático", "componentes" y "funcionamiento".

Los estudiantes presentan dificultades en la apropiación de "concepto matemático" y "funcionamiento". La figura 22 indica en el tercer nivel con color rojo la mayor dificultad la cual se basa en la "comprensión del concepto", y la menor dificultad con color verde correspondiente a la identificación de los componentes de la máquina de Turing.

Lo anterior se reconoce mejor con la expresión verbal de uno de los entrevistados:

> Desde mi punto de vista y desde mi experiencia la que realizamos pues una máquina de Turing con una cinta unidimensional, para mí lo que puede ser lo más difícil de comprender vendría siendo el algoritmo para realizar una acción de cómo se llega a crear ese algoritmo y la lógica que se hace tras ese algoritmo. (Participante 2)

La lógica es inherente al cerebro humano, el funcionamiento de la máquina es mecánico, unir estos dos mundos es lo abstracto y lo complejo, de ahí la dificultad en la comprensión del concepto formal de la máquina de Turing.

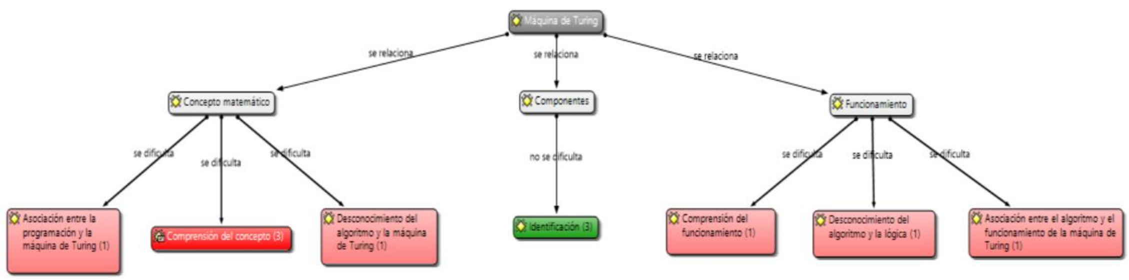
> Se cree que la idea de que el procedimiento de decisión es un procedimiento mecánico, es decir puede ser ejecutado por un ente sin inteligencia, siempre y cuando tenga acceso a las instrucciones indicadas por el (estás sí creadas con inteligencia) llevo a Turing a concebir la idea de una máquina

abstracta para describir este concepto. Esta máquina abstracta se conoce actualmente como la máquina de Turing y corresponde a la noción del concepto de algoritmo. (Sicard, 1996, p.31)

Respecto a sus componentes, la identificación de sus partes fue clara para el grupo, debido a que es estructural su funcionalidad en conjunto.

> Es importante resaltar el genio del Sr. Turing al poder haber desarrollado una Máquina tan compleja y con tantas aplicaciones prácticas que perduran en la actualidad, con componentes sencillos y una lógica sensacional logro convertirse un uno de los padres de la computación y la informática (Vilca, 2016, p.39)

Figura 22. Dificultades de apropiación variable máquina de Turing



Fuente: construcción de la autora.

4.2.1.2 Funciones recursivas

La variable "funciones recursivas" es la que más subvariables contiene y son: "concepto de relación", "concepto de función", "tipo de funciones", "función computable", y "concepto de función recursiva". La figura 23 presenta en el tercer nivel la mayor dificultad en color rojo, correspondiente a: "asociación entre el proceso matemático y la fórmula" en la subvariable "función computable", mientras que la menor dificultad está en color verde.

Las dificultades evidenciadas en términos generales provienen de la comprensión en conceptos matemáticos y la asociación con fórmulas y/o aplicaciones propias de la matemática discreta; como lo evidencia uno de los entrevistados, haciendo referencia a las dificultades de "función recursiva".

> En mi caso digamos que fue complicado entenderlo los conceptos y relacionarlos sin embargo cuando hice el ejemplo de las cajas más o menos entendí, sin embargo, es complejo porque más o menos uno cree que es como si se llamara a sí misma y se tiende a relacionar todos estos conceptos a veces algunos están bien y otros no están bien entonces se dificulta porque no hemos aprendido a relacionar unos temas con otros (Participante 3)

La apropiación del conocimiento involucra relacionar las teorías vistas y la manera de conjugarlas para así estructurar mejor el conocimiento, alrededor de la variable funciones recursivas se observa un denominador común la asociación y relación entre estos.

> Los trabajos que involucran funciones resultan problematicos para estudiantes de diferentes niveles, sin embargo, brindan una gran oportunidad para explorar diferentes representaciones del mismo objeto en un mismo ambiente, lo que facilita su estudio. Por tanto hablar de representación equivale a hablar de conocimiento, comprensión, modelización. (Amaya y Medina, 2013, p.119).

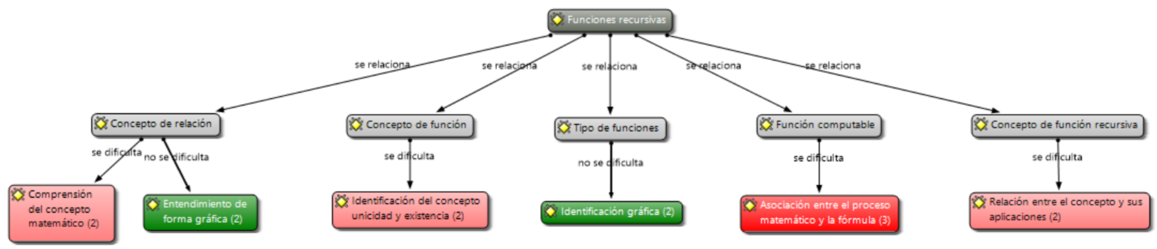
La variable "funciones recursivas" al ser uno de los ejes más importantes del estudio de computabilidad, tiene un número amplio de aplicaciones en la ingeniería de sistemas; sin embargo, al establecer su nexo con la teoría no lo relacionaban en un primer momento; pero, al realizar una presentación gráfica mejoró la comprensión del tema, evidenciando una mejor apropiación.

Trejos menciona la importancia de las diversas presentaciones del conocimiento

> Desde el punto de vista académico, toda búsqueda de nuevas formas de solución algorítmica a problemas tradicionales abre puertas para que el pensamiento, la lógica, la algoritmia, la programación y las matemáticas converjan para que el estudiante encuentre un espacio tanto de significado como de descubrimiento. (Trejos, 2015, p.20).

Además, en relación con el valor significativo de la aplicación en la ingeniería Trejos (2015) "Siempre es posible encontrar un camino para que las ciencias básicas, mayormente representadas por las matemáticas, encuentren un significado que facilite su aprendizaje y su aplicación" (p.28).

Figura 23. Dificultades de apropiación variable funciones recursivas



Fuente: construcción de la autora.

4.2.1.3 Lógica computable

La tercera variable en el análisis cualitativo es la dimensión de la lógica en la computabilidad, denominada "lógica computable"; para el caso de la "matemática discreta" se representa en las siguientes subvariables: "lógica proposicional", "circuitos lógicos" y "teoría de conjuntos".

Los circuitos lógicos son una forma de aproximación al tema de la lógica, el cual es abstracto. La complejidad de los dispositivos tecnológicos nace de la sencillez y la materialidad de las compuertas: AND, OR, NOT, NAND, NOR, XOR.

A través de una protoboard se hace el acercamiento a la "matemática discreta", desde el interés propio del estudiante de ingeniería de sistemas, aunado a la interpretación gráfica obteniendo la menor dificultad en el aprendizaje como se observa en la figura 24 en color verde.

Cuando se relaciona la lógica aristotélica con la práctica de las compuertas según lo expresa Freund (2011) "El conceptualismo puede jugar un papel importante en la resolución de problemas de filosofía de las matemáticas o de la lógica, ha sido empleado también como un fundamento para el desarrollo de teorías matemáticas y lógicas." (p.10) Por lo anterior la relación entre las diferentes teorías contribuye a una mejor comprensión y una menor dificultad en el conocimiento como lo expone uno de los entrevistados:

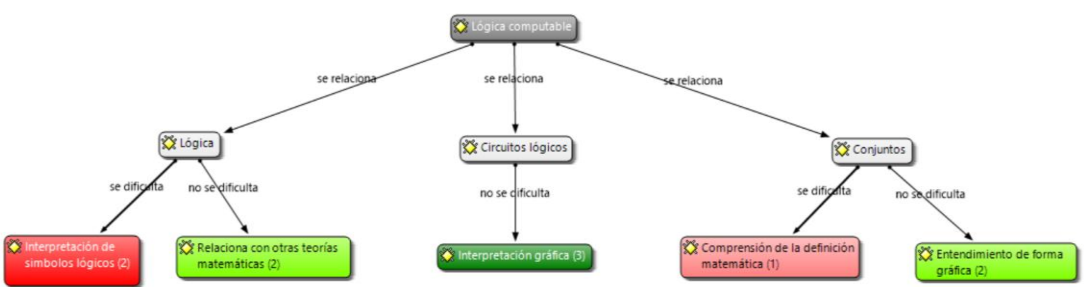
> Pues para mí es un tema que con la práctica no resulta siendo complejo es un poco de memorizar de por si es un tema muy interesante de cómo se relaciona la lógica proposicional que se veía en los temas de filosofía y todo esto con lo que vendría siendo conjuntos, álgebra booleana y las compuertas y todo eso. Para mí no es un tema muy complejo por el interés que le di también. (participante 2)

Como se observa en el tercer nivel del recuadro verde de la primera subvariable en la figura 24. La mayor dificultad se evidenció en la interpretación de los símbolos lógicos, por ello está en color rojo. Brouwer el fundador del intuicionismo expresa:

> La comunicación de las ideas es una función del lenguaje matemático, pero está herramienta es imperfecta. El pensamiento matemático que es estricto y uniforme en sí mismo, se vuelve susceptible de oscuridad y de error cuando es transferido de una persona a otra por medio del habla o la escritura (Brouwer, 1975, p.15).

Esto conlleva a ser la didáctica la herramienta más adecuada para entender los temas más abstractos de la matemática. Debido a que el camino recorrido en la lógica computable es de lo más abstracto.

Figura 24. Dificultades de apropiación variable lógica computable



Fuente: construcción de la autora.

4.2.1.4 Algoritmia

La algoritmia es fundamental en la formación disciplinar de la ingeniería de sistemas, por ello dicha temática hace parte del diseño curricular desde los primeros semestres y es una de las cuatro variables de este trabajo de investigación.

En matemática discreta, “la algoritmia” se aborda desde las siguientes tres subvariables: “construcción de algoritmos”, “características de los algoritmos” y “diagramas de flujo”.

La mayor dificultad, tal como se muestra en la figura 25, en color rojo corresponde a: “conocimientos básicos de los diagramas de flujo”; mientras que la menor dificultad se encuentra en color verde, asociado a la “interpretación gráfica de un algoritmo” mediante un diagrama de flujo, los cuales expone Barrera (2013) “Utilizar diagramas para representar un algoritmo tiene claras ventajas desde la perspectiva didáctica. Investigaciones han mostrado que el aprendizaje visual es uno de los mejores métodos para enseñar habilidades del pensamiento.” (p.4)

Las dificultades en la variable algoritmia son las menos subsanables en el sentido del tiempo curricular asignado en “matemática discreta”; la identificación de las características propias de un diagrama de flujo estuvo muy marcada en el análisis cualitativo, por ello se requiere un esfuerzo integral de otras asignaturas como, “programación de computadores”.

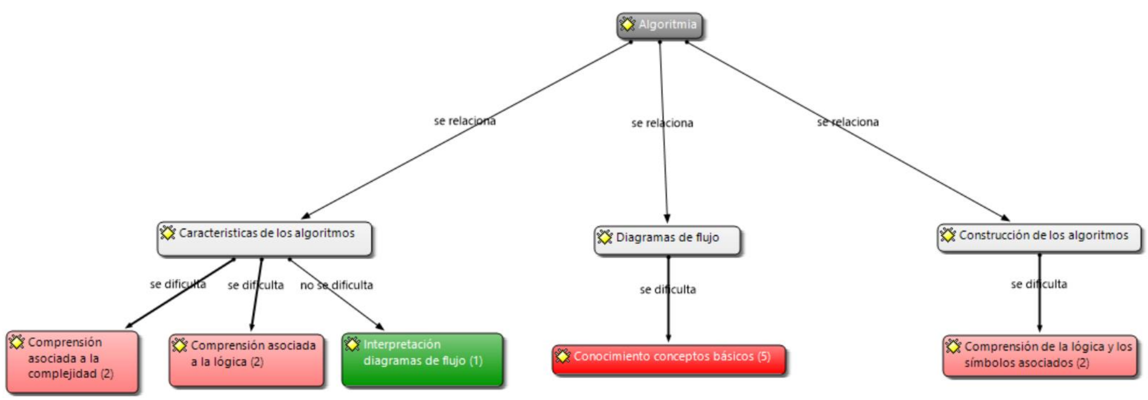
En general los estudiantes expresan que existe dificultades asociados a la complejidad, a la lógica y a la simbología.

> La comprensión de un algoritmo depende mucho de la complejidad de este, no todos los algoritmos son fáciles de entender, un algoritmo para sumar un número no es lo mismo que un algoritmo para identificar un numero o si un número es computable cada algoritmo tiene su complejidad y su estudio para entenderlo y depende mucho de esto. (Participante 3).

Se reitera la necesidad de la formación algorítmica en estudiantes de ingeniería de sistemas, los cuales están en primera línea frente a los retos tecnológicos del siglo XXI;

> En una sociedad definida como "del conocimiento", los procesos del pensamiento requeridos para enfrentar las actuales tareas han cambiado y adquieren relevancia nuevas formas de pensar. En este contexto, las habilidades del pensamiento procedimental utilizadas en la programación de computadores y ligadas al pensamiento algorítmico aparecen como una oportunidad para el desarrollo de las destrezas para resolver problemas en la Matemática" (Barrera, 2013, p.3)

Figura 25. Dificultades de apropiación variable algoritmia



Fuente: construcción de la autora.

4.3 Niveles de compresión y dificultades asociadas

El trabajo se realiza utilizando el modelo mixto, mediante la asociación de datos de diferente naturaleza.

La computabilidad además de ser compleja, la captura de dicho concepto no ha sido fácil, y no termina de construirse; por otro lado, la dificultad en la apropiación por parte de los estudiantes amerita la búsqueda, una de las razones de ser de este trabajo investigativo; donde se integra lo mejor del método cuantitativo y el método cualitativo mediante un diseño secuencial explicativo.

De acuerdo con Hernández et al. (2014), "el enfoque cuantitativo establece que el conocimiento debe ser objetivo y deductivo" (p.573); la matemática de la computabilidad es un buen referente de esta caracterización. A diferencia del análisis cualitativo, la entrevista se realizó de manera inductiva, con la voz propia de cada estudiante, tratando la complejidad de la enseñanza – aprendizaje, utilizando la comunicación de las nuevas plataformas como Teams.

Se establecen cuatro matrices correspondientes a cada una de las variables identificadas: "máquina de Turing", "funciones recursivas", "lógica computable" y "algoritmia"; la posición de las variables se encuentra a lo largo de la primera columna en cada una de las matrices.

En la segunda columna de cada matriz, se ubican las subvariables; las cuales tienen negrilla y en sombra gris para diferenciarlas fácilmente. Cada una de estas subvariables tiene la dimensión del análisis cualitativo que pretenden evidenciar las dificultades del aprendizaje; los componentes cualitativos de cada subvariable se encuentran en la segunda columna debajo de la respectiva subvariable.

Los colores rojo, rosado y verde hacen referencia al grado de dificultad en la apropiación del concepto; el rojo representa una mayor dificultad, el rosado una dificultad media y el verde la menor dificultad. Cada componente tiene la pregunta asociada al cuestionario cuantitativo que presentaron los estudiantes identificado con la letra "P" mayúscula, seguida del número de la pregunta.

El resultado del análisis cuantitativo de cada variable se encuentra en la parte derecha de la tabla encabezando las últimas tres columnas en sentido horizontal de cada matriz donde se diferencia el nivel bajo, del nivel medio y alto de acuerdo con el respectivo porcentaje de asertividad, el cual cambia en cada una de las cuatro variables.

Lo importante de cada matriz son los resultados asociativos entre el análisis cuantitativo y cualitativo; estos resultados, se encuentran en la parte inferior derecha de cada matriz, zona que es identificable por los signos más y menos en cada una de las cuatro matrices.

Primera matriz: Variable "Máquina de Turing"

La primera matriz (Tabla 12) se aprecia la correspondencia mediante los símbolos más y menos entre los resultados cuantitativos presentados en la figura 17 y las dificultades en la voz de los estudiantes indicado en la figura 22, correspondiente al análisis cualitativo.

La correspondencia entre el bajo nivel del 81% con la apropiación de la "comprensión del concepto matemático de la máquina de Turing", reflejan la relación entre los dos análisis. Además, las dificultades identificadas en las subvariables concepto matemático y funcionamiento, siendo la más relevante comprensión del concepto, son determinantes en el resultado de un nivel bajo de apropiación en la variable máquina de Turing.

Es relevante la identificación de los componentes de la "máquina de Turing": (cinta, cabezal o scanner y el programa) que tuvo por un lado la menor dificultad de acuerdo con los resultados cualitativos, y el mayor asertividad de acuerdo con los resultados cuantitativos, con un alto nivel de apropiación, contribuyendo a un nivel medio de apropiación de la variable.

Tabla 12.

Matriz de análisis mixto – variable máquina de Turing

Variable		Máquina de Turing		
		Análisis Cuantitativo		
	Subvariable/dificultades	Niveles		
	Análisis Cualitativo	Bajo: 81%	Medio: 15%	Alto: 4%
Máquina de Turing	**Concepto matemático**			
	Asociación entre programación y la máquina de Turing (P1)	-	+	-
	Comprensión del concepto (P3)	+	-	-
	Desconocimiento del algoritmo (P2)	+	-	-
	Componentes			
	Identificación (P.4, P.5, P.6)	-	+	-
	Funcionamiento			
	Comprensión del funcionamiento (P7, P.8)	+	-	-
	Desconocimiento del algoritmo y la lógica (P.9)	-	+	-
	Asociación entre el algoritmo y el funcionamiento de la máquina (P.10)	+	-	-

Fuente: creación propia

Segunda matriz: Variable "Funciones Recursivas"

Respecto a la segunda matriz Tabla 13 "Funciones Recursivas", es importante anotar como a partir de la visión mixta se puede obtener una inferencia que surge más desde el análisis cualitativo, ya que los estudiantes expresan que la "dimensión gráfica" hace bastante didáctica la enseñanza de los temas; esto es muy relevante porque la variable "Funciones Recursivas" requiere para su tratamiento del más alto nivel de formalización matemática, obteniendo desde el análisis cuantitativo un nivel medio y alto en su apropiación.

Tabla 13.

Matriz de análisis mixto – variable Funciones Recursivas

Variable		**Funciones Recursivas**		
		Análisis Cuantitativo		
	Subvariables/dificultades	N i v e l e s		
	Análisis Cualitativo	**Bajo: 23%**	**Medio: 54 %**	**Alto: 23 %**
Funciones recursivas	**Concepto de relación**			
	Comprensión del concepto matemático (P.11)	-	+/-	-
	Entendimiento de forma gráfica (P.12)	-	+	+/-
	Concepto de función			
	Identificación del concepto unicidad y existencia (P.13)	-	+/-	+
	Tipo de funciones			
	Identificación gráfica (P.15, P.16, P.17)	-	-	+
	Función computable			
	Asociación entre el proceso matemático y la fórmula (P.18)	+	-	-
	Concepto de función recursiva			
	Relación entre el concepto y sus aplicaciones (P.20, P.21)	+	-	-

Fuente: creación propia

En la subvariable "función computable" los estudiantes expresaron que su mayor dificultad corresponde a la asociación entre la dimensión matemática y la simbolización algebraica lo cual se correlaciona con el análisis cuantitativo donde obtuvieron un nivel de apropiación bajo.

Lo anterior es el resultado de relacionar los datos cuantitativos figura 18 y el análisis cualitativo figura 23.

Tercera matriz: Variable "Lógica computable"

La matriz "lógica computable" Tabla 14, tiene tres subvariables: lógica, circuitos lógicos y teoría de conjuntos, las cuales forman un isomorfismo que permite abordar dicha temática, desde los circuitos lógicos que es el tema más comprensible para el estudiante

Es recurrente que se encuentra la menor dificultad, en la comprensión de los modelos gráficos que representan el simbolismo y la axiomática, por ello en este tipo de preguntas se obtuvo un nivel alto de apropiación el cual se representa en color verde. El análisis mixto se obtuvo en la correlación de los resultados obtenidos en el análisis cuantitativo figura 19 y los cualitativos figura 24.

Tabla 14.

Matriz de análisis mixto – variable Lógica computable

Variable		**Lógica computable**		
		Análisis Cuantitativo		
	Subvariables/dificultades	Niveles		
	A. Cualitativo	Bajo: 11%	Medio: 31%	Alto: 58%
Lógica computable	**Lógica proposicional**			
	Interpretación de símbolos lógicos (P.23)	+	+/-	-
	Relaciona con otras teorías matemáticas (P.22)	-	-	+
	Circuitos lógicos			
	Interpretación gráfica (P.26)	-	-	+
	Conjuntos			
	Comprensión de la identificación matemática (P. 30)	-	+	-
	Entendimiento de la forma gráfica (P.28)	-	+/-	+

Fuente: creación propia

Cuarta matriz: Variable “Algoritmia”

La matriz algoritmia tabla 15, se construyó mediante la correlación de los datos obtenidos en el método cuantitativo figura 20 y el método cualitativo figura 25, la cual tiene tres subvariables, características de los algoritmos, diagrama de flujo y construcción de un algoritmo.

La menor dificultad se encuentra en interpretación de diagramas de flujo, donde las preguntas asociadas a la prueba de conocimiento fue la número 33, donde el porcentaje de asertividad es cercano al 96%. Evidenciando el carácter de modelación que se requiere en algoritmia para la solución de un problema. La mayor dificultad se asocia con los conocimientos de los conceptos básicos en la estructuración de los diagramas de flujo, obteniendo un 57% de asertividad en las preguntas asociadas al tema en la prueba de conocimiento, lo cual concuerda con el análisis cualitativo donde expresaron los entrevistados tener vacíos en esta temática.

Tabla 15.

Matriz de análisis mixto - variable algoritmia

Variable		**Lógica computable**		
		Análisis Cuantitativo		
	Subvariables/dificultades	Niveles		
	Análisis Cualitativo	Bajo: 4%	Medio: 27%	Alto: 69%
	Características de los algoritmos			
	Comprensión asociada a la complejidad (P.32)	-	+/-	+
	Comprensión asociada a la lógica (P.31)	-	-	+
Algoritmia	Interpretación diagramas de flujo (P.33)	-	-	+
	Diagramas de flujo			
	Conocimiento conceptos básicos (P.35)	+/-	+	-
	Construcción de los algoritmos			
	Comprensión de la lógica y los símbolos asociados (P.36)	+	+/-	-

Fuente: creación propia

Capítulo 5. Conclusiones

La computabilidad entendida desde la ingeniería de sistemas es un concepto estructurante y transversal tanto en la formación curricular como en las aplicaciones de las múltiples ramas de la tecnología. Es importante resaltar que si bien la computabilidad es propia de la ingeniería de sistemas, esta se utiliza en todas las ingenierías y en muchas actividades del ser humano.

La computabilidad, se hizo explicita desde la matemática en la década del 30 del siglo pasado a partir del trabajo de los pioneros como Turing, Church, Post, Gödel, Kleene y Markov entre otros; esta aventura tecnológica tomó un impulso que incidió transversalmente el siglo XX; la formalización que tanto importaba a Hilbert tuvo sus consecuencias y realidades que llegan hasta nuestros días y pareciera ser la impronta del siglo XXI.

Es dentro de este marco que se inscribe esta investigación, más precisamente en el contexto de la formación científico-técnica en jóvenes estudiantes que optaron por la tecnología como su formación profesional; de ahí la importancia en hacer una aproximación a las dificultades asociadas a la apropiación del concepto de "computabilidad" y este trabajo muestra que es posible hacer de la computabilidad una temática en los primeros semestres y desde la arista de la "matemática discreta" y no solamente con las materias que tradicionalmente se ha hecho, tales como "programación de computadoras".

En este capítulo se describen los hallazgos asociados con los objetivos de investigación, también se presentan las nuevas ideas que surgen como producto de este trabajo, las limitantes y preguntas para futuras investigaciones.

5.1 Principales hallazgos

¿Cuáles son los niveles de comprensión en la apropiación del concepto de computabilidad en estudiantes de Ingeniería de Sistemas?

Desde el punto de vista cuantitativo, los niveles de apropiación obtuvieron en general una media del 62%, sin discriminar el género; sin embargo, es importante resaltar que la brecha de género femenino es del orden del 19%, la cual es inferior a la referenciada en otros estudios en áreas de ciencia y tecnología que son del 32%; sin embargo, el género femenino presentó un nivel de apropiación alto del 25%, frente a los hombres de un 17%, este dato es relevante en cuanto que se concibe el campo de tecnología como un área propia del género masculino.

Con respecto a las cuatro variables establecidas en el estudio, la “máquina de Turing” fue la determinante para tener nivel bajo en la puntuación global. En nivel medio están las dos variables “funciones recursivas” y “lógica computable” en su orden y en nivel alto la variable “algoritmia” fue la que mejor contribuyó a la puntuación general.

Con los resultados cuantitativos obtenidos se hace necesario profundizar en el curso frente a las variables “máquina de Turing” y “funciones recursivas” debido a que son las variables que presentaron un nivel de apropiación bajo del 81% y 23% respectivamente; estas son temáticas fundamentales y transversales en el que hacer de la ingeniería de sistemas.

¿Cuáles son las dificultades asociadas a la apropiación del concepto de computabilidad en los estudiantes de Ingeniería de Sistemas?

Las principales dificultades de la apropiación del concepto de “computabilidad” provienen de “la matemática de la computabilidad”, y específicamente tienen que ver con la comprensión de los conceptos matemáticos, el simbolismo y la algebrización.

Respecto a la primera variable máquina de Turing la principal dificultad es el alto nivel de abstracción para comprender la funcionalidad en su conjunto; en cuanto a la variable funciones recursivas, la mayor dificultad proviene del concepto matemático de función, relación, recursividad, biyección que son fundamentales para entender la computabilidad; en la dimensión de la lógica aristotélica, correspondiente a la tercera variable lógica computable las dificultades surgieron en la interpretación y algebrización de los símbolos lógicos; la última variable algoritmia presento dificultades en los temas asociados a las características propias en la representación de los diagramas de flujo, que se encuentran más relacionados con las nuevas formas de pensar en el modelo algorítmico.

Lo anterior, es la mirada desde cada una de las cuatro variables estructurantes del concepto de computabilidad; ahora, desde un enfoque global se requiere una visión holística del isomorfismo existente entre la teoría de conjuntos, la lógica proposicional, los circuitos lógicos y el álgebra booleana.

¿Cómo se asocian los niveles de apropiación del concepto de "computabilidad" y las dificultades que experimentan los estudiantes del curso de «matemáticas discretas» del programa de Ingeniería de Sistemas en la Escuela Tecnológica Instituto Técnico Central – ETITC?

En la primer variable "máquina de Turing" se presentó la mayor dificultad, estableciéndose una correspondencia entre el bajo nivel de apropiación con respecto a las dificultades que experimentaron los estudiantes con la conceptualización matemática de dicha variable; respecto a la variable "funciones recursivas" se hace evidente la dificultad debido al alto nivel de formalización matemática que requiere el tratamiento de estos temas; en la tercer variable "lógica computable", el mayor nivel de dificultad se presentó con la interpretación de los símbolos lógicos;

por último, en la cuarta variable: “algoritmia”, se evidenció las dificultades relacionadas con la estructuración de los diagramas de flujo en cuanto al pensamiento algorítmico.

En general, los resultados muestran que las dificultades caracterizadas son próximas a los niveles de apropiación (identificados) alrededor del concepto de “computabilidad”.

5.2 Generación de nuevas ideas

¿Qué debería considerarse en la enseñanza de la “computabilidad” a estudiantes de ingeniería de sistemas?

Los desarrollos en ciencia y la tecnología exigen una constante valoración en su dimensión social; en este caso, se pretende tener unos nuevos lineamientos para el diseño curricular en la ingeniería de sistemas y es parte de lo que se hizo en este trabajo; mostrar que es posible la enseñanza de la computabilidad desde los primeros semestres con nuevos enfoques y no solamente desde la perspectiva tradicional de las materias a las cuales se les ha dejado este rol, los cursos correspondientes a los “lenguajes de programación”; los cuales no está por demás decir que si bien son necesarios, son insuficientes para tener toda la perspectiva de lo que hoy significa la “computabilidad”.

En particular, se abordó el estudio desde la “matemática discreta”, en temas como “teoría de conjuntos” donde es importante introducir conceptos como “conjunto computable”; en el tema de funciones, “funciones computables”, y la enseñanza de la “máquina de Turing”, temática que aparece en los diseños curriculares de ingeniería de sistemas en la mayoría de las universidades en los últimos semestres.

Enseñar el “modelo de Turing” en estudiantes de primeros semestres, no es solo importante por la definición misma de “computabilidad”, es también recuperar la visión sintética de Alan Turing, quien reduce la complejidad de la computabilidad a un modelo “mecánico operacional”.

Turing dejó mucho por hacer, pero en estos 80 años de recorrido el camino por él demarcado ha demostrado su eficacia, pertinencia y es necesario expresar que a pesar de los muchos intentos, el modelo de Turing no ha sido superado.

El esfuerzo de la revolución industrial del siglo XVIII consistió en trasladar el "músculo humano" a la máquina; ahora en el siglo XXI el "modelo de Turing" es trasladar parte de las funciones del "cerebro humano" a la máquina en lo que se ha dado en llamar "inteligencia artificial"; por ello desde la introducción se planteó que "la computabilidad" a pesar de aparecer recientemente no es nueva en el desarrollo histórico social de la humanidad; hoy más que nunca es necesario mantener el espíritu de Turing, hacer aprehensible la complejidad que requiere hoy la realidad.

A partir del estudio se infiere que para la enseñanza de la computabilidad se recomienda seguir los lineamientos planteados a continuación:

1. Es posible enseñar la computabilidad desde los primeros semestres con el enfoque desde la "matemática discreta", como se propone en este trabajo de investigación y no esperar hasta los últimos semestres de formación en estudiantes de ingeniería para abordar la importancia y el entendimiento de esta temática.

2. Comenzar desde temáticas donde los estudiantes tienen fortalezas, en concordancia con su formación en la educación media como por ejemplo la "teoría de conjuntos".

3. Fomentar el trabajo creativo en la realización de temas muy relacionados como circuitos lógicos.

4. Profundizar la enseñanza mediante el modelo gráfico.

5. Asumir la enseñanza de la "computabilidad" con modelo como el aprendizaje de la "máquina de Turing", realizar esto desde los primeros semestres contribuye a la formación del

estudiante de ingeniería de sistemas. Se presentan algunos ejemplos realizados en semestres anteriores por los estudiantes en el curso de Matemáticas discretas de la ETITC:

Link 1 https://www.youtube.com/watch?v=FUv9ievAMPA

Link 2 https://www.youtube.com/watch?v=Txue3lAeMgQ

6. Se propone una secuencia didáctica para la enseñanza de la “matemática discreta” estructurada en las siguientes temáticas: (a) Teoría de conjuntos, (b) Conjuntos computables, (c) Compuertas lógicas, (d) Máquina de Turing, (e) Lógica proposicional, (f) Lógica computable, (g) Funciones recursivas, (h) Funciones computables, (i) Algoritmia. Los ítems d.) y g.) son los temas en los cuales se debe profundizar, con más horas de enseñanza en teoría y práctica debido a que fue en estos tópicos donde se presentaron los niveles más bajos de apropiación (Método cuantitativo) y en la voz de los estudiantes donde se presentaron las mayores dificultades en la apropiación (Método cualitativo).

7. Romper el compartimento desde cada una de estas temáticas es muy importante, una manera de hacerlo es profundizar en el isomorfismo entre lógica proposicional, teoría de conjuntos, circuitos lógicos y álgebra Booleana, ello permite rehacer didácticamente la enseñanza aprendizaje de una manera más adecuada.

8. La impronta de la revolución tecnológica en el siglo XXI requiere deconstruir la *caja negra* de los dispositivos, la formación científico - técnica en los jóvenes estudiantes que optan por la tecnología como su formación profesional requiere de este esfuerzo y el país lo necesita para no ser simples consumidores en un mundo globalizado.

5.3 Limitantes

1. El concepto de "computabilidad" tiene altos niveles de complejidad, no ha sido fácil en la historia de la humanidad lograr lo que se ha hecha hasta ahora, y es claro que este concepto no termina de construirse.
2. Otro limitante importante además del anterior, es lo correspondiente a las dificultades propias de la enseñanza del concepto de computabilidad.
3. Normalmente se considera que "la computabilidad" solo puede ser abordada desde las clásicas materias de "programación de computadores"; las cuales son necesarias para el aprendizaje de los "lenguajes de programación", pero son insuficientes para abordar la dimensión de la computabilidad.
4. El universo muestral estuvo conformado por 26 estudiantes, y este tipo de trabajo es la primera vez que se realiza en la ETITC; se propone por lo tanto desarrollar este tipo de estudios en próximos semestres para validar o refutar los resultados obtenidos.
5. La muestra del grupo focal que se estableció para el análisis cualitativo fue de 6 estudiantes, para próximos trabajos se propone aumentar el número de estudiantes. En el momento de la realización de la entrevista asistieron dos estudiantes nivel alto, dos nivel medio y uno nivel bajo, con ello la entrevista se realizó a 5 estudiantes. Debido a que el estudiante que no asistió obtuvo un nivel bajo de apropiación si es relevante no haber podido conocer sus dificultades para tener un mejor análisis.
6. Dentro del marco generado por la crisis del COVID-19 de una formación eminentemente presencial se pasó a realizar las clases de forma remota; dicho cambio fue demasiado rápido, sin preparación por parte de los profesores y estudiantes porque tocó asumir la nueva realidad de la cuarentena y el confinamiento.

7. Existen pocos estudios en la dimensión realizada en este trabajo; el “statuo quo” en la academia y en los programas curriculares hace que se instauren formas y contenidos que permanecen en el tiempo, cuando la realidad es que se requieren cambios urgentes tanto en contenido como en la forma de enseñanza.

5.4 Nuevas preguntas de investigación

Existen diversas preguntas que surgen en esta misma línea, relacionadas con el tema de investigación, estas son:

1. ¿Cómo determinar la pertinencia del currículo de ingeniería de sistemas en términos de contenidos y objetivos en la materia “matemáticas discretas” frente a la realidad tan cambiante de la tecnología en el siglo XXI?
2. ¿Cuáles son las nuevas didácticas que está utilizando la academia, los docentes y el rol del estudiante en la “educación remota” que se impone actualmente?
3. ¿Cuál es el papel de la asignatura “Matemáticas Discretas” en concordancia con los nuevos retos y desarrollos de la computabilidad?
4. ¿Cuáles son las oportunidades de mejoramiento que podrían tener los currículos de la materia “matemáticas discretas”?

Referencias

Aguiar, B. & Gutiérrez, P. & Lara, B. & Villalpando, B. (2011). El rendimiento académico de las mujeres en matemáticas: análisis bibliográfico y un estudio de caso en educación superior en México. *Revista Electrónica Actualidades Investigativas en Educación*, 11(2),1-24. Recuperado de https://www.redalyc.org/articulo.oa?id=447/44720020016

Amaya, T., & Medina, A. (2013). Dificultades de los estudiantes de grado once al hacer transformaciones de representación de una función con el registro como registro principal. Educación matemática, 25 (2). pp 119 – 140.

Alonso, L. (2013). Turing su tesis doctoral. *Revista Investigación y Ciencia*. No. 437. pp 94 – 95. Disponible en https://www.investigacionyciencia.es/files/12194.pdf

Badallo, A., González, O., Orellana, R., & Montero, D. (2018). *Cultura científica y cultura tecnológica: Actas del IV Congreso Iberoamericano de Filosofía de la Ciencia y la Tecnología* (Vol. 250). Ediciones Universidad de Salamanca.

Barchini, G. (2008). Las Máquinas de Turing como modelo general de la computación. ¿Hacia un cambio de paradigma? *Revista Colombiana de Computación.* V10, No.1. pp 7-25. Recuperado de https://revistas.unab.edu.co/index.php/rcc/article/download/1134/1104/

Barrera, L (2013). Algoritmos y programación para la enseñanza y aprendizaje de la matemática escolar. *VII CIBEM*. Recuperado de: http://funes.uniandes.edu.co/18563/1/Barrera2013Algoritmos.pdf

Blanco, J. & Ludeña, C. (2015). *Matemáticas Discretas.* Escuela de Computación. Universidad Central de Venezuela. Recuperado de http://www.ciens.ucv.ve/discretasiii/Apuntes/md03.pdf

Boole, G. (1848). *The Mathematical Analysis of Logic, Being an Essay towards a Calculus of Deductive Reasoning.* Macmillan: London.

Bourque, P., & Fairley, R. (2014). *SWEBOK Guide to the Software Engineering Body of Knowledge*, Version 3.0. IEEE Computer Society. Recuperado de: www.swebok.org

Brousseau, G. (1986). Fondements et méthodes de la didactiques des mathématiques. *Recherches en didactique des Mathématiques.* 7(2), 33-115.

Bruner, J. (1987). *La importancia de la educación*. Paidos.

Brouwer, L (1975), *Collected Works.* Ámsterdam, Holanda, North–Holland Publishing.

Castro, I. (2011). *Máquinas de Post. Apuntes de Clase.* Bogotá: Editorial Universidad Nacional de Colombia.

Church, A. (1936). An unsolvable problem of elementary number theory. *American Journal of Mathematics*. 58, pp. 345 – 363.

Copeland, B. J., Posy, C. J., & Shagrir, O. (2013). *Computability: Turing, Gödel, Church, and beyond.* New York: The MIT Press.

Davis, Martin. (1965). *The Undecidable.* Heweltt, New York: Raver Press.

Davis Martin. (2000). *La computadora Universal de Leibniz a Turing.* ISBN 84-8306-953-9

De Camilloni, A., Davini, M. C., Edelstein, G., Litwin, E., Souto, M., & Barco, S. (1998). *Corrientes didácticas contemporáneas.* Paidós.

Dedekind R. (1872). Continuity and irrational numbers.

ETITC. (2020). Syllabus Matemática Discreta. Currículo de Ingeniería de Sistemas. Recuperado de www.etic.edu.co.

Epp, S. (2011). *Discrete Mathematics with Applications*. Boston: Brooks/Cole.

Fernández, G. & Saez, F. (1987). *Fundamentos de informática.* Madrid: Alianza.

Freudenthal, H. (1986). *Fenomenología didáctica de las estructuras matemáticas.* Springer. ISBN 9789027722614.

Freund, M. (2011). Lógica, matemáticas y conceptualismo. *Signos filosóficos*, *13*(25), 9-45. Recuperado de http://www.scielo.org.mx/scielo.php?script-=sci_arttext&pid=S1665-13242011000100001&lng=es&tlng=es.

Godel, K. (2006). *Obras completas de Kurt Godel. Vol I, Vol II, Vol III.* Traducción e Introducción de Jesús Mosterín. Madrid: Alianza Editorial.

Guzmán, M. (2007). Enseñanza de las Ciencias y la Matemática. *Revista Iberoamericana de Educación* No. 43. pp 19 – 58. Recuperado de https://rieoei.org/historico/documentos/rie43a02.pdf

Hilbert, D. (1993). *Fundamentos de las matemáticas*. México: Editorial Servicios editoriales de la facultad de ciencias UNAM. Colección MATHEMA.

Hilbert, D. & Ackermann, W. (1928). *Grundge der theoretischen logik*. Berlin. Julius Springer.

Hodges, A. (1983). *Alan Turing: The enigma.* New York: Simon and Schuster.

Hawking, S. (2006). *Dios creo los números. Los descubrimientos matemáticos que cambiaron la historia*. Ed Critica-es.

Hernández, R, & Mendoza, C (2018). *Metodología de la investigación*. Las rutas cuantitativa, cualitativa y mixta, Ciudad de México, México: Editorial Mc Graw Hill.

Hernández, R., Fernández, C., & Baptista, P. (2014). *Metodología de la investigación*. México D.F.: McGraw-Hill.

Johnsonbaugh, R. (2005). *Matemáticas Discretas.* Chicago: Pearson.

Kemo, S. (2019). Digital 2019: Global internet use accelerates. Recuperado de https://wearesocial.com/blog/2019/01/digital-2019-global-internet-use-accelerates

Leavitt, D. (2006). *El hombre que sabía demasiado. Alan Turing y la invención de la computadora.* Barcelona: Antoniobosch.

Martinez, T. (2010). *Taller de programación orientada a objetos.* Recuperado de: http://gemmatmtz.blogspot.com/2010/11/calculo-lambda_10.html

McMillan., J y Sally., S (2005): *Investigación Educativa*. Una introducción conceptual. Madrid: Pearson Addison Wesley.

Mota, S. (2018). Dos concepciones del lenguaje: Wittgenstein y Chomsky en torno a la recursión como “buena” explicación de la naturaleza humana. *Praxis Filosófica.* (46) 125-149.

Patiño, L. (2020). Una maratón en tacones. Recuperado de https://www.eltiempo.com/tecnosfera/novedades-tecnologia/brecha-de-genero-mujeres-estudian-ciencia-y-tecnologia-en-colombia-412134

Penrose, R. (1989). *La mente nueva del emperador. En torno a la cibernética, la mente y las leyes de la física*. New York: Oxford University Press.

Quevedo, J. (2009). *Decidability and Constructivism.* IEEE Computer Society, 1711-1715.

Ramirez, J. (2017). Matemáticas Discretas I - G2 - Código 2025963. Universidad Nacional de Colombia – Sede Bogotá. Profesor: José Luis Ramírez. Segundo Semestre 2017. Contenido de la materia Recuperado de https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxjdXJzb3Nqb3NlbHJhbWlyZXp8Z3g6N2IxMzlhZTg4MDhkZTg0NA

Rodríguez, J. (2013) Una mirada a la pedagogía tradicional y humanista, *Presencia Universitaria*, 3(5) pp. 36-4. Recuperado de: http://eprints.uanl.mx/3681/1/Una_mirada_a_la_pedagog%C3%ADa_tradicional__y_hum anista.pdf

Rosen, K. (2004). *Matemática Discreta y sus aplicaciones*. New York. Ma Graw Hill

Rosenfeld, R., & Irazábal, J (2013) *Computabilidad, complejidad computacional y verificación de programas*. La plata: Universidad Nacional de la Plata.

Seymour Lipschutz, S. y Lipson, M. (2009). *Matemáticas Discretas.* Virginia USA. Ma Graw Hill

Shurkin, J. (1984). *Engines of the mind: a history of the computer*. Editorial W.W. Norton & Co. ISBN 10: 0393018040

Sicard, A (1996). Máquinas de Turing. *Revista universidad EAFIT*. 32(103).

Soare, R. (1996). *The history and concept of computability*. University of Chicago. Recuperado de http://www.people.cs.uchicago.edu/~soare/History/handbook.pdf.

Solar, H., García, B., Rojas, F., & Coronado, A. (2014). Propuesta de un Modelo de Competencia Matemática como articulador entre el currículo, la formación de profesores y el aprendizaje de los estudiantes. *Educación Matemática*, 26(2), 33-67. Recuperado de https://www.redalyc.org/articulo.oa?id=405/40532665003

Todd, Z., Nerlich, B., & McKeown, S. (2004). Introduction. En Z. Todd, B. Nerlich, S. McKeown & D. Clarke (Eds.), *Mixing methods in psychology*. 3-16. Hove, East Sussex, UK: Psychology Press

Torreti, R. (1998). *El Paraiso de Cantor. La tradición conjuntista en la filosofía matemática*. Editorial Universitaria. Chile.

Trejos, O. (2015). Algoritmo Recursivo diferente para hallar elementos de la serie de Fibonacci usando Programación Funcional. *Avances Investigación en Ingeniería*. Vol. 11 - No. 2 ISSN: 1794-4953

Trejos, O. (2017). Metodología algorítmica para construir funciones que resuelvan cálculos basados en procesos simples usando programación funcional. *Investigación en ingeniería* DOI: doi.org/10.1804/avances.v14i1 63-76

Turing, A. (1936). On Computable Numbers, with application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*. 2(42), 230-265. https://doi.org/10.1112/plms/s2-42.1.230

Turing, A. (1937). On Computable Numbers, with an Application to the Entscheidungsproblem: A correction. *Proceedings of the London Mathematical Society*. 2(43), 544-546.

Vergnaud, G. (1990). *Teoría de los campos conceptuales*. Editorial Madrid.

Vilca, C. (2016). *Máquinas de Turing y sus aplicaciones.* ISSN 2415-2323

Wearsocial. (22 de marzo 2020), *Digital around the world in 2020*. [imagen]

Wearsocial. (22 de marzo 2020), *Internet user numbers over time* [imagen]

Xu, Zhi-Wei & Tu, Dan-Dan. (2011). Three new concepts of future computer science. *Journal of Computer Science and Technology*. Beijing Tomo 26, N.º 4, (Jul 2011): 616-624. http://dx.doi.org/10.1007/s11390-011-1161-4 ID del documento de ProQuest875935835

APÉNDICES

Apéndice A. Consentimiento informado

APLICACIÓN INSTRUMENTOS DE RECOLECCIÓN DE DATOS

Yo,______________________ con documento de identidad________________ en mi calidad de estudiante, autorizo de manera voluntaria, libre y espontánea para aplicar los instrumentos de recolección de datos: del cuestionario sobre computabilidad del trabajo de investigación titulado: "Dificultades en la apropiación del concepto de computabilidad", cuyo objetivo es: determinar niveles alto, medio y bajo de comprensión en la apropiación del concepto de computabilidad en los estudiantes del curso Matemáticas Discretas, del programa Ingeniería de sistemas.

La duración estimada de esta prueba es de 120 minutos. Agradecemos su valiosa participación.

Se firma en la ciudad de________________________a los_días el mes de ________________ de 2020.

Atentamente,

(Nombre completo y Firma)

Apéndice B. Matriz de categorías

Objetivos de investigación	Categoría	Sub categoría	Instrumento
Determinar los niveles de apropiación del concepto de "computabilidad" en los estudiantes del curso "matemáticas discretas" del programa de Ingeniería de Sistemas.	Máquina de Turing	Concepto matemático de una máquina de Turing	Cuestionario
		Componentes de una máquina de Turing	
		Funcionamiento de una máquina de Turing	
	Funciones Recursivas	Concepto de relación	
		Concepto de función	
		Tipo de funciones	
		Función computable	
		Concepto de función recursiva	
	Lógica Computable	Lógica	
		Circuitos lógicos	
		Conjuntos	
	Algoritmia	Características de un algoritmo	
		Características diagrama de flujo	
		Construcción de un algoritmo	

Objetivos de investigación	Categoría	Sub categoría	Instrumento
Describir las dificultades asociadas a la apropiación del concepto de "computabilidad", a partir del reconocimiento de la voz de los estudiantes del curso "Matemáticas discretas" del programa de Ingeniería de Sistemas	Máquina de Turing	Concepto matemático de una máquina de Turing	Entrevista grupal
		Componentes de una máquina de Turing	
		Funcionamiento de una máquina de Turing	
	Funciones Recursivas	Concepto de relación	
		Concepto de función	
		Tipo de funciones	
		Función computable	
		Concepto de función recursiva	
	Lógica computable	Lógica	
		Circuitos lógicos	
		Conjuntos	
	Algoritmia	Características de un algoritmo	
		Características diagrama de flujo	
		Construcción de un algoritmo	

Apéndice C. Instrumentos de recolección de datos

VARIABLE	SUBVARIABLE	PREGUNTAS ASOCIADAS
Máquina de Turing **(V.1)** Es un dispositivo que manipula símbolos sobre una tira de cinta de acuerdo con una tabla de reglas. A pesar de su simplicidad, una máquina de Turing puede ser adaptada para simular la lógica de cualquier algoritmo de computador y es particularmente útil en la explicación de las funciones dentro de un computador.	**Concepto matemático de una máquina de Turing.** **(V.1.1)** Es un dispositivo que manipula símbolos sobre una tira de cinta de acuerdo con una tabla de reglas. A pesar de su simplicidad, una máquina de Turing puede ser adaptada para simular la lógica de cualquier algoritmo de computador y es particularmente útil en la explicación de las funciones dentro de un computad	**(V.1.1.1)**. Seleccione la opción que considere correcta. Qué se puede afirmar del modelo de computabilidad de la ***«Máquina de Turing»***: en relación con su funcionalidad en el siglo XXI. a. Es un modelo que por ser realizado en el año 1936 (hace 84 años) ya es obsoleto. b. Es un modelo muy simple y sencillo de entender y no ha sido superado. c. Es un modelo que debido el desarrollo tecnológico del siglo XXI poco se utiliza. d. Es un modelo muy complejo y difícil de entender. **(V.1.1.2).** Seleccione la opción que considere correcta. Para la ingeniería de sistemas la computabilidad como propiedad atribuible a cierta clase de objetos es: a. Una propiedad más como cualquier otra. b. Una propiedad que tiene carácter binario excluyente. c. Una propiedad que tiene carácter binario incluyente. d. Ninguna de las anteriores. **(V.1.1.3).** Seleccione la opción que representa la forma correcta de operar una Maquina de Turing: a. Inicialización de la Máquina / Introducir la cadena de entrada en la cinta / La cinta está en blanco para iniciar / Se posiciona la cabeza sobre la primera letra de la cadena introducida / Se pone la unidad de control en el estado inicial. b. Inicialización de la Máquina / Introducir la cadena de entrada en la cinta / La cinta está en blanco para iniciar / Se pone la unidad de control en el estado inicial / Se posiciona la cabeza sobre la primera letra de la cadena introducida. c. Inicialización de la Máquina / La cinta está en blanco / Introducir la cadena de entrada en la cinta / Se pone la unidad de control en el estado inicial / Se posiciona la cabeza sobre la primera letra de la cadena introducida. d. Inicialización de la Máquina / Se pone la unidad de control en el estado inicial / La cinta está en blanco / Introducir la cadena de entrada en la cinta / Se posiciona la cabeza sobre la primera letra de la cadena introducida

<table>
<tr>
<td></td>
<td>Componentes de una máquina de Turing.

(V.1.2)

Los componentes de una máquina de Turing son: La cinta, el cabezal o scanner, y el programa.</td>
<td>(V.1.2.1). Una «Máquina de Turing» se define usando n-tuplas.
¿Cuál es el valor de n?.

Seleccione la opción que considere correcta

a. n =8
b. n =4
c. n =7
d. Ninguna de las anteriores

(V.1.2.2). La cinta es uno de los cuatro componentes de la «Máquina de Turing». En el paper de Turing, la cinta es:

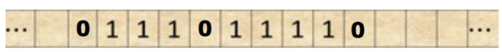
Seleccione la opción que considere correcta.

a. Unidimensional.
b. Bidimensional.
c. Tridimensional.
d. Ninguna de las anteriores.

(V.1.2.3). ¿Cuál de los cuatro componentes de la máquina de Turing se asocia al software de un computador? Seleccione la opción que considere correcta

en la «Máquina de Turing»?

a. En la cinta
b. En el cabezal o scanner.
c. En el Registro de Estado.
d. En la Tabla de Instrucciones</td>
</tr>
</table>

	Funcionamiento de una máquina de Turing. **(V.1.3)** Es un sistema descrito por una tabla de máquina que contiene una lista de estados y una serie de instrucciones. El sistema posee los medios necesarios para identificar las entradas, emitir las salidas y cambiar de estado. La tabla de máquina define los estados únicamente en términos funcionales.	**(V.1.3.1).** Seleccione la opción que considere correcta. La cinta es uno de los cuatro componentes de la ***«Máquina de Turing»***, la cinta se divide en cuadros y cada cuadro recibe el nombre de celda, cada celda está en blanco o tiene un símbolo, una sola y única marca. En determinado momento el cabezal o scanner lee una celda de la cinta. Se puede afirmar que: a. Los símbolos en otros lugares de la cinta afectan el comportamiento de la máquina. b. Los símbolos en otros lugares a la izquierda pueden afectar el comportamiento de la máquina. c. Los símbolos en otros lugares de la cinta no afectan el comportamiento de la máquina. d. Los símbolos en otros lugares a veces afectan el comportamiento de la máquina. **(V.1.3.2).** Seleccione la opción que considere correcta. La ***«Máquina de Turing»*** modela matemáticamente una máquina que opera mecánicamente sobre una cinta que se divide en celdas una al lado de la otra. En esta cinta hay símbolos que la máquina puede: a. Leer. b. Escribir. c. Borrar. d. Todas las anteriores. **(V.1.3.3)**. Seleccione la opción que considere correcta. Si en una ***«Máquina de Turing»*** tiene el siguiente ***input*** en la cinta:  Este ***input*** representa computablemente: a. La suma de varios 0's y 1's b. La suma de 0+1+1+1+0 c. La suma de 3 + 4 d. Ninguna de las anteriores **(V.1.3.4).** Con respecto al "output" de ***«Máquina de Turing»***. ¿Qué puede suceder con respecto a los posibles resultados: ***output*** ? 1. Puede producir una respuesta (output). 2. Puede que nunca se detenga. Seleccione la opción que considere correcta. a. Solo puede darse la opción 1. b. Solo puede darse la opción 2. c. Puede darse cualquiera de las opciones 1 y 2. d. Ninguna de las anteriores.

<table>
<tr>
<td rowspan="3">Funciones Recursivas
(V.2)

Una función es recursiva si, para cada n-tupla $\langle \boldsymbol{x_1}, \dots \boldsymbol{x_n} \rangle$ de números naturales se puede derivar de $\boldsymbol{E}$, conforme a las reglas de derivación, a lo sumo para un numeral $\boldsymbol{X}$ una ecuación de la forma $\boldsymbol{f}(\boldsymbol{X_1}, \dots, \boldsymbol{X_n}) = \boldsymbol{X}$, donde $\boldsymbol{X_{1,\dots,}} \boldsymbol{X_n}$ son los numerales.</td>
<td>Concepto de relación
(V.2.1)

Una relación es una correspondencia entre dos elementos de dos conjuntos con ciertas propiedades.</td>
<td>(V.2.1.1). Indique si la siguiente afirmación es falsa o verdadera:

Toda función es relación.

a) Verdadero
b) Falso

(V.2.1.2). Seleccione la opción que considere correcta:
Al segundo par ordenado de una relación se le conoce como el:_________

a) Discriminante
b) Rango
c) Contra dominio
d) Dominio</td>
</tr>
<tr>
<td>Concepto de función
(V.2.2)

Es una relación entre dos conjuntos de tal forma que a cada elemento del dominio le corresponde un único elemento del codominio.</td>
<td>(V.2.2.1). Seleccione la opción que considere correcta.

¿Cuáles son las condiciones para que una relación sea función?

a. Que la relación sea inyectiva y sobreyectiva
b. Que la relación cumpla con unicidad y existencia
c. Que la relación sea biyectiva e inyectiva
d. Todas las anteriores

(V.2.2.2). Indique si la siguiente afirmación es falsa o verdadera.
Toda relación es función

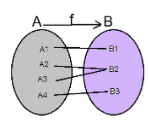
a) Verdadero
b) Falso</td>
</tr>
<tr>
<td>Tipo de funciones

(V.2.3)

Función inyectiva. Es aquella en la que a elementos distintos el dominio le corresponden elementos distintos del recorrido.

Función sobreyectiva. Es aquella en la que a todo elemento del recorrido le corresponde cuando menos un elemento del dominio.

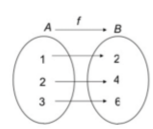
Función biyectiva. Es aquella que es inyectiva y sobreyectiva.</td>
<td>(V.2.3.1). Seleccione la opción que considere correcta.
Analice el siguiente gráfico.



Teniendo en cuenta la gráfica anterior cuál de las siguientes definiciones de función representa.

a. Inyectiva
b. Sobreyectiva
c. Biyectiva
d. Ninguna de las anteriores

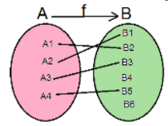
(V.2.3.2). Seleccione la opción que considere correcta.
Analice el siguiente gráfico.



Teniendo en cuenta la gráfica anterior cuál de las siguientes definiciones de función representa.

a. Inyectiva
b. Sobreyectiva
c. Biyectiva
d. Ninguna de las anteriores

(V.2.3.3). Seleccione la opción que considere correcta.
Analice el siguiente gráfico.</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td>

Teniendo en cuenta la gráfica anterior cuál de las siguientes definiciones de función representa.
a. Inyectiva
b. Sobreyectiva
c. Biyectiva
d. Ninguna de las anteriores </td>
</tr>
<tr>
<td></td>
<td> Función computable
(V.2.4)
Las funciones computables son las funciones que pueden ser calculadas por una máquina de Turing. </td>
<td> (V.2.4.1). Seleccione la opción que considere correcta.
El conjunto de las funciones computables es______________
a. Numerable
b. Aritmética
c. Booleana
d. Recursiva
(V.2.4.2). Indique si la siguiente afirmación es falsa o verdadera.
Todo número racional es computable
a. Verdadero
b. Falso </td>
</tr>
<tr>
<td></td>
<td> Concepto de función recursiva
(V.2.5)
Es una función que se invoca así misma y es un proceso que se repite hasta que llega a un punto de solución. Lo cual debe ser expresado en el algoritmo. </td>
<td> (V.2.5.1). Seleccione la opción que considere correcta.
Qué condiciones se deben cumplir para que una función sea recursiva:
a. Vuelve a invocar al algoritmo
b. Existe una salida no recursiva del procedimiento
c. Su caso base no vuelva a invocarse
d. Un conjunto de vectores entrelazados
(V.2.5.2). Seleccione la opción que considere correcta.
De las siguientes curvas cual cumple con el concepto de función recursiva
a. Curva de Peano
b. Curva de Hilbert
c. Curva de Koch
d. Todas las anteriores </td>
</tr>
<tr>
<td> Lógica computable
(V.3)
La computabilidad es el estudio matemático de los modelos de computación. Además la teoría de la computabilidad es la parte de la computación que estudia los problemas de decisión que se pueden resolver con un algoritmo o equivalentemente con una máquina de Turing </td>
<td> Lógica
(V.3.1)
Es la forma del razonamiento, por medio de reglas y técnicas determina si un argumento es válido. Trata de métodos de razonamiento. </td>
<td> (V.3.1.1). Poner una X en la casilla correspondiente de la tabla del SI o NO.
El desarrollo de la computabilidad, puede establecerse desde la teoría de conjuntos de Cantor, cuando dos o más conjuntos se interrelacionan su resultado, su computabilidad produce un conjunto resultado a partir de su operabilidad.
Definición:
Un conjunto $\boldsymbol{A} \subseteq \mathbf{N}$ es computable si existe un programa que, $\forall \mathbf{n}$, decide si $\mathbf{n} \in \mathbf{A}$.
Cuando tenemos un algoritmo que determina si se cumple o no una propiedad, el conjunto o relación definido por ella se llama decidible. Así pues, el concepto intuitivo de computabilidad efectiva para funciones se generaliza para conjuntos.
<table>
<tr><th>Conjunto</th><th>SI</th><th>NO</th></tr>
<tr><td>El conjunto de números pares es un conjunto computable</td><td>x</td><td></td></tr>
</table>
(V.3.1.2). Poner una X en la casilla correspondiente de la tabla del SI o NO. </td>
</tr>
</table>

El desarrollo de la computabilidad, puede establecerse desde la teoría de conjuntos de Cantor, cuando dos o más conjuntos se interrelacionan su resultado, ***su computabilidad*** produce un conjunto resultado a partir de su operabilidad.

Definición:
Un conjunto $\boldsymbol{A} \subseteq \mathbf{N}$ ***es computable*** si existe un programa que, $\forall$**n**, decide si $\mathbf{n} \in \mathbf{A}$.

Cuando tenemos un algoritmo que determina si se cumple o no una propiedad, el conjunto o relación definido por ella se llama decidible. Así pues, el concepto intuitivo de computabilidad efectiva para funciones se generaliza para conjuntos.

Conjunto	**SI**	**NO**
El conjunto de números primos es un conjunto computable	x	

(V.3.1.3). Poner una X en la casilla correspondiente de la tabla del SI o NO.

El desarrollo de la computabilidad, puede establecerse desde la teoría de conjuntos de Cantor, cuando dos o más conjuntos se interrelacionan su resultado, ***su computabilidad*** produce un conjunto resultado a partir de su operabilidad.

Definición:
Un conjunto $\boldsymbol{A} \subseteq \mathbf{N}$ ***es computable*** si existe un programa que, $\forall$**n**, decide si $\mathbf{n} \in \mathbf{A}$.

Cuando tenemos un algoritmo que determina si se cumple o no una propiedad, el conjunto o relación definido por ella se llama decidible. Así pues, el concepto intuitivo de computabilidad efectiva para funciones se generaliza para conjuntos.

Conjunto	**SI**	**NO**
El conjunto de los programas escritos correctamente es un conjunto computable	x	

(V.3.1.4). Seleccione una de las siguientes respuestas posibles:

Con respecto a la lógica de proposiciones se puede afirmar que. Seleccione una (1) de las siguientes opciones.

a. Es posible determinar intuitivamente el valor de verdad de una proposición compuesta.

b. Una tabla de verdad describe el valor de verdad que puede tomar una proposición compuesta para todas las combinaciones de los valores de verdad de sus proposiciones simples.

c. El valor de verdad de una proposición compuesta depende del valor de verdad de sus proposiciones simples y está no cambia al cambiar el conectivo lógico.

d. Si una proposición compuesta tiene dos proposiciones simples y un conectivo lógico, existirán ocho combinaciones posibles de los valores de verdad de las premisas.

<table>
<tr>
<td></td>
<td>

Circuitos lógicos

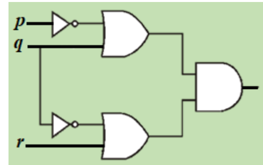
(V.3.2)

Son estructuras formales (sistemas abstractos) que representan sistemas para transmisión de información de toda índole. Los circuitos lógicas están compuestos por elementos digitales como las compuertas AND, OR y NOT.

</td>
<td>

(V.3.2.1). Seleccione la opción que considere correcta.

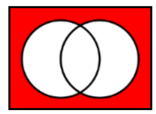
La representación del siguiente circuito lógico:



Esta dada por:

a. (~p v ~q) ∧ (~q v r)
b. (~p v q) ∧ (~q v ~r)
c. (~p v q) ∧ (~q v r)
d. (~p v q) ∧ ~ (~q v r)

(V.3.2.2). Analice el siguiente gráfico de teoría de conjuntos



¿Qué compuerta lógica representa el área roja? Seleccione la opción que considere correcta

a. AND
b. NAND
c. OR
d. XOR

</td>
</tr>
<tr>
<td></td>
<td>

Conjuntos

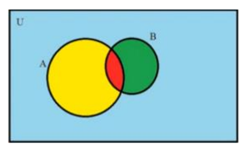
(V.3.3)

Un conjunto es una colección de elementos con características similares considerada en sí misma como un objeto. Además, es una
colección de elementos, pertenecientes a la misma categoría, y cuya agrupación puede ser considerada o identificada en sí misma como un objeto.

</td>
<td>

(V.3.3.1). Analice el siguiente gráfico de teoría de conjuntos

De acuerdo al siguiente gráfico



¿Qué color representa el color amarillo? Seleccione la opción que considere correcta

a. A-B
b. B-A
c. A∪B
d. A∩B

(V.3.3.2). Dados los conjuntos A y B, encontrar la operación que genera el conjunto C. Donde el conjunto universal U es:

U = {Juan, Ana, Carlos, Diego, María}
A={Juan,Ana}
B = {María, Carlos}
C = {Diego}

Seleccione la opción que considere correcta.

a. A U B’
b. (A U B)’
c. U – B
d. A’

</td>
</tr>
</table>

		(V.3.3.3). Cuando en la asesoría de un grupo, el tutor solicita que levanten la mano los estudiantes que estudiaron el capítulo 1, pero no el capítulo 2. Es decir que levantarán la mano sólo los estudiantes que pertenezcan al conjunto de quienes estudiaron el capítulo 1, pero que a su vez no hayan leído ningún tema del capítulo 2. Teniendo en cuenta el enunciado anterior, seleccione la opción que representa la operación. a. Diferencia b. Unión c. Intersección e. Complemento
Algoritmia **(V.4)** Es un conjunto ordenado y finito de operaciones que permite encontrar la solución a un problema cualquiera.	**Características de un algoritmo** **(V.4.1)** Las características son: primero se modela el problema que se necesita resolver, a continuación se diseña la solución, luego ésta se analiza para determinar su grado de corrección y eficiencia, y finalmente se traduce a instrucciones de un lenguaje de programación que un computador entenderá	**(V.4.1.1).** Indique si la siguiente afirmación es falsa o verdadera. **En un algoritmo se deben contemplar y declarar las variables a usar** a. Verdadero b. Falso **(V.4.1.2).** Seleccione la opción que considere correcta. **¿Qué es un algoritmo?** a. Un programa escrito en un lenguaje de programación específico. b. Conjunto ordenado y finito de operaciones para resolver un problema concreto. c. Un operador numérico que realiza un proceso determinado. d. Conjunto ordenado e infinito de operaciones para resolver un problema concreto. **(V.4.1.3).** Seleccione la opción que considere correcta. **Son dos características de los algoritmos:** a. Corto y rápido b. Finito y preciso c. Rápido y correcto d. Infinito y preciso
	Características diagrama de flujo **(V.4.2)** Un diagrama de flujo es una representación gráfica de un proceso. Cada paso del proceso es representado por un símbolo diferente que contiene una breve descripción de la etapa de proceso. Los símbolos gráficos del flujo del proceso están unidos entre sí con flechas que indican la dirección de flujo del proceso	**(V.4.2.1).** Indique si la siguiente afirmación es falsa o verdadera. **Un diagrama de flujo es la forma de representar un algoritmo por medio de gráficos que representan objetos o cajas de texto** a. Verdadero b. Falso **(V.4.2.2).** Seleccione la opción que considere correcta. **¿Qué es un diagrama de flujo?** a. Es la representación gráfica de un algoritmo. b. Es un programa gráfico que muestra un proceso. c. Es la representación numérica de un algoritmo. d. Es un programa numérico que muestra un proceso. **(V.4.2.3).** Seleccione la opción que considere correcta. **De las siguientes cuatro opciones. ¿Cuál utiliza símbolos para identificar las instrucciones a realizar dentro de un programa**? a. Algoritmo b. Proceso c. Diagrama de flujo

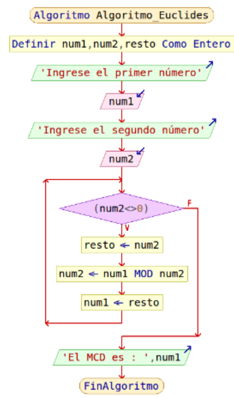
d. Objetos

Construcción de un algoritmo

(V.4.3)

Todo algoritmo debe obedecer la estructura básica de un sistema o los elementos del problema es decir, datos de entrada, proceso y datos de salida.

(V.4.3.1). Evalué el siguiente "Diagrama de Flujo" Denominado "Algoritmo de Euclides"



¿Considera que este algoritmo está correctamente diseñado?

a. SI
b. NO

(V.4.3.2). Evalué el siguiente algoritmo de las "Torres de Hanoi"

Entrada: Tres pilas de números *origen*, *auxiliar*, *destino*, con la pila origen ordenada

Salida: La pila destino

1. **sí** origen =={1} **entonces**
1. **mover** el disco 1 de pila origen a la pila destino (insertarlo arriba de la pila destino)
2. **terminar**
2. **si no**
Hanoi ({*1, … ,n-1*}, *origen*, *destino*, *auxiliar*)
//mover todas las fichas menos la más grande (*n*) a la varilla auxiliar
3. **mover** disco *n* a destino //mover la ficha grande hasta la varilla final
4. hanoi (*auxiliar*, *origen*, *destino*)
//mover todas las fichas restantes, *1*...*n–1*, encima de la ficha grande (*n*)
5. **terminar**

¿Considera que este algoritmo está correctamente diseñado?

a. SI
b. NO

VARIABLE	SUBVARIABLE	PREGUNTA ASOCIADA
Máquina de Turing **(F.V.1)**	Concepto matemático de una máquina de Turing. **(F.V.1.1)**	**(F.V.1.1.1)** ¿En términos generales, es fácil o difícil entender qué es la Máquina de Turing?
	Componentes de una máquina de Turing. **(F.V.1.2)**	**(F.V.1.2.1)** ¿Cuáles son las dificultades para identificar las partes de la máquina de Turing?
	Funcionamiento de una máquina de Turing. **(F.V.1.3)**	¿Qué es lo más difícil de comprender en el funcionamiento de una máquina de Turing? **(F.V.1.3.1)**
Funciones Recursivas **(F.V.2)**	Concepto de relación **(F.V.2.1)**	**(F.V.2.1.1)** ¿Es fácil de comprender el concepto de relación?
	Concepto de función **(F.V.2.2)**	**(F.V.2.2.1)** ¿Qué dificultades tiene al identificar una función?
	Tipo de funciones **(F.V.2.3)**	**(F.V.2.3.1)** ¿Qué criterios siguen para clasificar una función?
	Función computable **(F.V.2.3)**	**(F.V.2.4.1)** ¿Qué dificultades encontró al identificar una función computable?
	Concepto de función recursiva **(F.V.2.5)**	**(F.V.2.5.1)** ¿Cuáles son los aspectos que dificultan comprender el concepto de función recursiva?
Lógica computable **(F.V.3)**	Lógica **(F.V.3.1)**	**(F.V.3.1.1)** ¿En términos generales, encuentran dificultades en la compresión de los conectores lógicos? ¿Cuáles?
	Circuitos lógicos **(F.V.3.2)**	**(F.V.3.2.1)** ¿Considera usted fácil o difícil la asociación de los circuitos lógicos y los conjuntos?
	Conjuntos **(F.V.3.3)**	**(F.V.3.3.1)** ¿Cómo identifica las operaciones entre conjuntos en diagramas de Veen?
Algoritmia **(F.V.4)**	Características de un algoritmo **(F.V.4.1)**	**(F.V.4.1.1)** ¿Consideran fácil o difícil la comprensión de un algoritmo? ¿Por qué?
	Características diagrama de flujo **(F.V.4.2)**	**(F.V.4.2.1)** ¿Qué aspectos se dificultan cuando se interpreta un diagrama de flujo?
	Construcción de un algoritmo **(F.V.4.3)**	**(F.V.4.3.1)** ¿Cuáles considera usted que son las dificultades para identificar si un algoritmo es correcto o no?

Protocolo grupo focal		
Título del proyecto:		Dificultades en la apropiación del concepto de computabilidad en estudiantes de ingeniería de sistemas.
Objetivo del grupo focal:		Identificar las dificultades asociadas a la apropiación del concepto de “computabilidad”, a partir del reconocimiento de la voz de los estudiantes del curso “Matemáticas discretas” del programa de Ingeniería de Sistemas.
Entrevistador:		Andrea García Rivas. Docente del curso Matemáticas discretas
No. grupo focal.	1	Participantes
1. Danna Valentina Coy García 2. Cesar Oswaldo Rodríguez Suarez 3. Nicol Valentina Alvis Buitrago 4. Fabio Orlin Alegría Vivero 5. Randy Joel Caballero López		

Fecha de aplicación:	Mayo 22 de 2020		Lugar:	Video conferencia – Software TEAMS	
Hora de inicio:	2:00 PM	Hora de finalización:	3:17 PM	Duración total:	1 hora 17 minutos

Presentación

La entrevista que hoy les realizare a ustedes cinco, estudiantes del curso Matemáticas Discretas asignatura que imparto en la Escuela Tecnológica Instituto Técnico Central. Actualmente llevo a cabo el trabajo de investigación titulado: “Dificultades en la apropiación del concepto de computabilidad” y tiene como uno de sus objetivos Identificar las dificultades asociadas a la apropiación del concepto de “computabilidad”, a partir del reconocimiento de la voz de los estudiantes del curso “Matemáticas discretas” del programa de Ingeniería de Sistemas. Por ello solicito a cada uno de ustedes de manera voluntaria, libre y espontánea su autorización para aplicar los instrumentos de recolección de información.

Cuestionario

1) ¿En términos generales, es fácil o difícil entender qué es la Máquina de Turing?
2) ¿Cuáles son las dificultades para identificar las partes de la máquina de Turing?
3) ¿Qué es lo más difícil de comprender en el funcionamiento de una máquina de Turing?
4) ¿Es fácil de comprender el concepto de relación?
5) ¿Qué dificultades tiene al identificar una función?
6) ¿Qué criterios siguen para clasificar una función?
7) ¿Qué dificultades encontró al identificar una función computable?
8) ¿Cuáles son los aspectos que dificultan comprender el concepto de función recursiva?
9) ¿En términos generales, encuentran dificultades en la compresión de los conectores lógicos? ¿Cuáles?
10) ¿Considera usted fácil o difícil la asociación de los circuitos lógicos y los conjuntos?
11) ¿Cómo identifica las operaciones entre conjuntos en diagramas de Veen?
12) ¿Consideran fácil o difícil la comprensión de un algoritmo? ¿Por qué?
13) ¿Qué aspectos se dificultan cuando se interpreta un diagrama de flujo?
14) ¿Cuáles considera usted que son las dificultades para identificar si un algoritmo es correcto o no?

Apéndice D. Constancia validación instrumentos identificación institucional

CONSTANCIA DE VALIDACIÓN

Yo, ***Eduardo Hernández Ortiz***, titular de la Cédula de Ciudadanía Nº 19.408.106 de Bogotá, de profesión Licenciado en administración y supervisión educativa con Maestría en comunicación educativa con énfasis en medios interactivos, ejerciendo actualmente como docente de educación superior de la facultad de Ingeniería de Sistemas, en la Institución Escuela Tecnológica Instituto Técnico Central.

Por medio de la presente hago constar que he revisado con fines de Validación del Instrumento (**Prueba de conocimiento**), a los efectos de su aplicación al personal que labora en Institución Escuela Tecnológica Instituto Técnico Central.

Luego de hacer las observaciones pertinentes, puedo formular las siguientes apreciaciones.

	DEFICIENTE	ACEPTABLE	BUENO	EXCELENTE
Congruencia de Ítems				X
Amplitud de contenido			X	
Redacción de los Ítems		X		
Claridad y precisión			X	
Pertinencia			X	

En Bogotá, a los 22 días del mes de abril del 2020

Eduardo Hernández O.

Firma

CONSTANCIA DE VALIDACIÓN

Yo, ***Eduardo Hernández Ortiz***, titular de la Cédula de Ciudadanía Nº 19.408.106 de Bogotá, de profesión Licenciado en administración y supervisión educativa con Maestría en comunicación educativa con énfasis en medios interactivos, ejerciendo actualmente como docente de educación superior de la facultad de Ingeniería de Sistemas, en la Institución Escuela Tecnológica Instituto Técnico Central.

Por medio de la presente hago constar que he revisado con fines de Validación del Instrumento (**cuestionario**), a los efectos de su aplicación al personal que labora en la Escuela Tecnológica Instituto Técnico Central.

Luego de hacer las observaciones pertinentes, puedo formular las siguientes apreciaciones.

	DEFICIENTE	ACEPTABLE	BUENO	EXCELENTE
Congruencia de Ítems				X
Amplitud de contenido			X	
Redacción de los Ítems		X		
Claridad y precisión		X		
Pertinencia			X	

En Bogotá, a los 22 días del mes de abril del 2020

Eduardo Hernández O.

Firma

IDENTIFICACIÓN INSTITUCIONAL

<u>CONSTANCIA DE VALIDACIÓN</u>

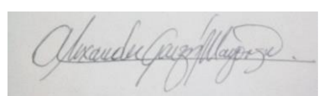
Yo, ***Alexander Garzón Mayorga***, titular de la Cédula de Ciudadanía Nº 1022369369 de Bogotá, de profesión Matemático con Maestría en Matemáticas, ejerciendo actualmente como consultor académico.

Por medio de la presente hago constar que he revisado con fines de Validación del Instrumento (**Prueba de conocimiento**), a los efectos de su aplicación al personal que labora en Institución Escuela Tecnológica Instituto Técnico Central.

Luego de hacer las observaciones pertinentes, puedo formular las siguientes apreciaciones.

	DEFICIENTE	ACEPTABLE	BUENO	EXCELENTE
Congruencia de Ítems				X
Amplitud de contenido			X	
Redacción de los Ítems			X	
Claridad y precisión			X	
Pertinencia				X

En Bogotá, a los 27 días del mes de abril del 2020



Firma

CONSTANCIA DE VALIDACIÓN

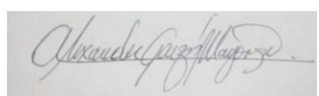
Yo, ***Alexander Garzón Mayorga***, titular de la Cédula de Ciudadanía Nº 1022369369 de Bogotá, de profesión Matemático con Maestría en Matemáticas, ejerciendo actualmente como consultor académico.

Por medio de la presente hago constar que he revisado con fines de Validación del Instrumento (**cuestionario**), a los efectos de su aplicación al personal que labora en la Escuela Tecnológica Instituto Técnico Central.

Luego de hacer las observaciones pertinentes, puedo formular las siguientes apreciaciones.

	DEFICIENTE	ACEPTABLE	BUENO	EXCELENTE
Congruencia de Ítems				X
Amplitud de contenido			X	
Redacción de los Ítems			X	
Claridad y precisión			X	
Pertinencia				X

En Bogotá, a los 27 días del mes de abril del 2020



Firma

Apéndice E. Transcripciones

Entrevista al grupo focal

1. ¿En términos generales, es fácil o difícil entender qué es la Máquina de Turing?

1. En mi opinión al principio se me hizo complicado entender, pero ya después se me facilito un poco mejor
2. En términos generales la máquina de Turing no es complicada, pero si queremos saber más a fondo de cómo es el funcionamiento es más complejo, pero en términos generales no es complicado
3.En mi opinión tampoco es complicado el concepto como tal de máquina y estoy de acuerdo con lo que dice Cesar (2)
4.En mi concepto tampoco es complicada
5.Yo también creo que no es complicada, si no que sería más fácil de entender si se tuvieran algunos conceptos básicos de programación como que es un numero binario.

2. ¿Cuáles son las dificultades para identificar las partes de la máquina de Turing?

En mi opinión ninguna pues cuando realizamos la máquina de Turing en casa no se me complico nada
1. Estoy de acuerdo con (3) las partes son muy claras para entenderlas
5. Yo también estoy de acuerdo

3. ¿Qué es lo más difícil de comprender en el funcionamiento de una máquina de Turing?

2. Desde mi punto de vista y desde mi experiencia la que realizamos pues una máquina de Turing con una cinta unidimensional, para mí lo que puede ser lo más difícil de comprender vendría siendo el algoritmo para realizar una acción de cómo se llega a crear ese algoritmo y la lógica que se hace tras ese algoritmo y tal vez como utilizar esto en una cinta bidimensional

4. ¿Es fácil de comprender el concepto de relación?

3. Personalmente se me complico en un inicio, pero relacionando esos conceptos por ejemplo el de reflexiva con el concepto de física que se ve en óptica se entiende mejor por decirlo así y el de transitividad se entiende y el de simetría por descarte se sabe cuál es, pero al principio se me complico bastante.
5. Yo creo que no es difícil a la hora de verlo de la manera que lo vimos no me pareció tan complicado o al menos a mí no se complicó tanto, pero si otra manera de aplicarlo a otro tipo de tema o algo que sea más real no lograría comprenderlo tan fácil

1. Yo pienso igual que el 5 no es tan complicado aparte de que la profe ayuda a entender

4. Al principio me pareció un poco complejo, pero ya lo comprendí un poco más.

5. ¿Qué dificultades tiene al identificar una función?

1. Yo al principio tuve problemas para identificar que es una función porque me enredaba mucho con lo de la unicidad y la existencia
4. Yo también un poco igual sobre la existencia y la unicidad porque o sea no sabía muy bien cómo identificarlo, pero lo principal es aprender el nombre y los conceptos para poder entenderlo
2. Desde mi perspectiva no es problemático si se tienen en cuenta las clasificaciones y diferenciar entre la unicidad y la existencia no se me hizo complicado teniendo en cuenta que lo aprendimos de una manera muy grafica

6. ¿Qué criterios siguen para clasificar una función?

3. En mi caso digamos que relacionado lo que dice 2 en grafica básicamente es en eso y en la inyectiva miro lo del conjunto del rango y la biyectiva sé que son ambas aparte de los conceptos también hago los gráficos
2. Para mi yo recordaba el concepto de existencia y unicidad y saber que era una función yo me guiaba por las flechas sabiendo si era inyectiva o sobreyectiva me guiaba por la orientación de las flechas y saber si le correspondían todos los elementos del rango del dominio y así, solo miraba las flechas

7. ¿Qué dificultades encontró al identificar una función computable?

1. En mi caso pues en algún momento leí que las funciones computables son un algoritmo que son funciones que pueden ser calculadas por alguna máquina, entonces más o menos me voy guiando y digamos que no es tan difícil identificar una función computable

4.Dificultades casi no profe porque cada función computable tiene su fórmula su regla entonces toca seguir la regla y así mismo entenderla

5.Lo que a mí se me dificulto un poco y no entiendo muy bien es como las computadoras logran hacer todos esos procesos matemáticos literalmente como saben sumar 4 más 4 pero el concepto de función computable lo entiendo, pero existen temas que aún no me quedan muy claros.

8. ¿Cuáles son los aspectos que dificultan comprender el concepto de función recursiva?

3. En mi caso digamos que fue complicado entenderlo los conceptos y relacionarlos sin embargo cuando hice el ejemplo de las cajas más o menos entendí, sin embargo, es complejo porque más o menos uno cree que es como si se llamara a sí misma y se tiende a relacionar todos estos conceptos a veces algunos están bien y otros o están bien entonces se dificultad porque no hemos aprendido a relacionar unos temas con otros
2. Para mi yo la verdad no conocía antes muy bien el concepto de funciones recursivas hasta que vi el cuestionario, sin embargo, recuerdo que tiene una utilidad en la programación y pues hice una especie de analogía cuando sabía que la función era recursiva era algo que se llamaba a sí misma como dijo 1. Por lo menos para mí no podría decirte muy bien lo que se me dificulta porque no entiendo realmente cómo funciona la función recursiva básicamente es eso

9. ¿En términos generales, encuentran dificultades en la compresión de los conectores lógicos? ¿Cuáles?

4. Al principio si tuve un poco de dificultad, pero ya cuando lo entendí que usted lo explico mejor si, aunque me confundía en el OR pero ya se me hace fácil. El and lo confundía con el or.
1. Pues yo al principio tuve como un poquito de complejidad, pero luego ya lo entendí perfecto por como la profe explica. Tuve complejidad ya cuando empezamos a ver lo de las cositas esa de tipos circuitos o algo así como cuando era el AND o el OR el que se utilizaba
2. Pues para mí es un tema que con la práctica no resulta siendo complejo es un poco de memorizar de por si es un tema muy interesante de cómo se relaciona la lógica proposicional que se veía en los temas de filosofía y todo esto con lo que vendría siendo conjuntos el álgebra booleana y las compuertas y todo eso. Para mí no es un tema muy complejo por el interés que le di también

10. ¿Considera usted fácil o difícil la asociación de los circuitos lógicos y los conjuntos?

3. En mi caso no creo lo que dice 2 es práctica y hacer ejercicios no se complica mucho
2. Para mí fue un tema nuevo y cuando lo estuve viendo me parecía un poquito intimidante pero no es muy complicado si se enfoca en las bases para entenderlo mejor, como saber cuál es la lógica tras de eso, tampoco hay mucha complejidad en eso.
1. En ese caso si fue sencillo de entender.

11. ¿Cómo identifica las operaciones entre conjuntos en diagramas de Veen?

1. Yo entendí más gráficamente
3. Yo entendía con la gráfica y con la definición matemática pues relacionada ambas cosas
2. Ambos se me hacían de utilidad yo usaba la definición matemática y usaba el diagrama de veen como para comprobarlo ambos me parecen buenas herramientas y me quedaría con ambas no solo con una.

12. ¿Consideran fácil o difícil la comprensión de un algoritmo? ¿Por qué?

5. Yo creo que es fácil porque los algoritmos lo característico es que son muy ordenados y creo que es fácil de entender

3.La comprensión de un algoritmo depende mucho de la complejidad del mismo, no todos los algoritmos son fáciles de entender, un algoritmo para sumar un número no es lo mismo que un algoritmo para identificar un numero o si un número es computable cada algoritmo tiene su complejidad y su estudio para entenderlo y depende mucho de esto

2. Entender el algoritmo como un concepto no es muy complicado, pero se sabe que es como una serie de pasos para llegar a resolver un problema, más o menos es así por encima un concepto general pero ya comprender un algoritmo depende de la lógica que haya usado la persona que creo al algoritmo yo diría que comprender esa lógica es importante para ya después entender el algoritmo y hay veces que el algoritmo se ve representado en diagramas de flujo y clases así, es un poco de uno involucrarse con eso, en conceptos generales no es complicado

4. Es un poco complejo, pero si uno lee entenderá mejor, yo hice algunos y al comienzo parece un poco duro, pero ahora ya me parece un poco más fácil

14. ¿Qué aspectos se dificultan cuando se interpreta un diagrama de flujo?

1. Yo pienso que se dificultad al interpretar un diagrama de flujo si uno no tiene los conceptos básicos claros

2. Yo también estoy de acuerdo con 1, yo veo un diagrama de flujo y a una primera impresión no lo entendería muy bien porque hay ciertas conexiones y cosas que tienen su significado

4. Yo estoy de acuerdo con ellos

2. Yo no lo entendí muy bien por qué además no estoy muy familiarizado en desarrollar o entender los diagramas de flujo, pues yo la verdad no lo entendí muy bien

3. Lo que pasa es que uno de los grupos no está familiarizado con el tema de diagramas de flujo y el otro si lo está, yo creo que en el campo de los diagramas de flujo siempre vamos a no entender porque no entendemos el concepto realmente

5. Si de hecho yo pertenecí a ese grupo y el profesor nos explicó con un programa sobre los diagramas de flujo y afortunadamente ya los había visto en el bachillerato, pues en mi caso se me facilita

15. ¿Cuáles considera usted que son las dificultades para identificar si un algoritmo es correcto o no?

3. Yo creo que gráficamente dependiendo de la simbología y de la lógica no sé si me hago entender muy bien ubicando esos dos un algoritmo por ejemplo vi el en cuestionario un algoritmo que tenía un dos mayor y un dos menor al tiempo entonces o es mayor o es menor igual y los símbolos no están ubicados lógicamente entonces uno dice esto aquí no cuadra o si de pronto hay una instrucción que no está haciendo algo útil.

2. Para mi si fue complicado entender no sabía muy bien lo que significaban algunas cosas y no deduje si funcionaba o no el algoritmo.

Currículum Vitae

Nombres: Andrea García Rivas

Identificación: 1032358944

Perfil:

Matemática e Ingeniera Química con especialización en Matemática aplicada, énfasis en sistemas dinámicos, con experiencia docente de cinco años en educación superior en la facultad de ingeniería de las instituciones Escuela tecnológica instituto técnico central (ETITC) y la Corporación Universitaria Republicana. He trabajo en el sector privado en el desarrollo de piezas de software específico del cálculo de bases de datos.

Experiencia:

Universidad de educación superior: Corporación Universitaria Republicana- Facultad de Ingeniería
Docente medio tiempo
Asignaturas: Álgebra Lineal, Lógica Matemática, Cálculo integral, Cálculo vectorial, Matemáticas discretas.

Universidad de educación superior: Escuela Tecnológica Instituto Técnico Central - Facultad de Ingeniería (ETITC)
Docente ocasional medio tiempo
Asignaturas: Álgebra Lineal, Matemáticas Básicas, Taller de Matemáticas, Geometría y Trigonometría, Matemáticas Discretas, Investigación de operaciones.

Encuentro Internacional de Ciencia, Tecnología e Innovación - México
Evaluador de proyectos científicos internacionales categoría ciencias exactas e ingeniería y computación. Conferencista [De Turing a los problemas Np]

Asesorías académicas Milton Ochoa (CEINFES)
Docente Pre-Saberes y Pre Universitarios

Grupo de Consultoría Informática (GCI)
Desarrollo de pruebas de piezas de software específico del cálculo de reservas matemáticas de ARL. Validaciones Matemáticas.

Universidad Nacional de Colombia
Monitora académica
Departamento de Ingeniería Química y Ambiental

Estudios:

Universidad Sergio Arboleda
Especialista en Matemática Aplicada con énfasis en Sistemas Dinámicos

Universidad Nacional de Colombia
Ingeniera Química

Universidad Distrital Francisco José de Caldas
Matemática

Instituto Pedagógico Arturo Ramírez Montufar (IPARM)
Universidad Nacional de Colombia
Bachiller académico