



**CARACTERIZACIÓN DEL PENSAMIENTO MATEMÁTICO DE ESTUDIANTES DE
SECUNDARIA, ESCUELA NORMAL SUPERIOR DE QUIBDÓ EN EL
DEPARTAMENTO DEL CHOCÓ, CUANDO RESUELVEN TAREAS MATEMÁTICAS
MEDIANTE PYTHON**

Kevin Estiguar Palacios Maturana

Tesis de Maestría en Educación

Asesor

Walter F. Castro G, PostDoctor (PostDoc) en Didáctica de las Matemáticas.

Universidad de Antioquia
Facultad de Educación
Maestría en Educación
Medellín, Antioquia, Colombia
2025

Cita	(Palacios Maturana, 2025)
Referencia	Palacios Maturana, K. E. (2025). Caracterización del Pensamiento Matemático de estudiantes de secundaria de la Escuela Normal Superior de Quibdó en el departamento del Chocó, cuando resuelven tareas matemáticas mediante Python [Tesis de maestría]. Universidad de Antioquia, Medellín, Colombia.
Estilo APA 7 (2020)	



Seleccione posgrado UdeA (A-Z), Cohorte XXIV.

Grupo de Investigación Matemática, Educación y Sociedad (MES).

Centro de Investigaciones Educativas y Pedagógicas (CIEP).



Centro de Documentación Educación

Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - www.udea.edu.co

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

DEDICATORIAS

A mis padres Margarita Maturana Maturana y Jesús Melquicedecs Palacios Mosquera, por ser los pilares de mi formación, por creer en mis capacidades, por su amor incondicional y por enseñarme que con esfuerzo y perseverancia se pueden alcanzar los sueños más ambiciosos. A ellos dedico este logro con todo mi amor y admiración.

AGRADECIMIENTOS

A mi asesor, Walter Fernando Castro Gordillo, por su dedicación, orientación constante, paciencia y compromiso con la excelencia académica durante todo el proceso investigativo.

A Emiro Borja Palacios, rector de la Escuela Normal Superior de Quibdó, por permitir la realización de este trabajo en la institución que también fue mi alma mater. A José Saucedo Machado, rector encargado, y a la coordinadora académica, por su colaboración y apertura durante la implementación de las actividades.

A los profesores Virna Lici Flórez Martínez (décimo grado), Jhon Jairo Chaverra (octavo grado) y Víctor Valencia (sexto grado), por permitirme trabajar con sus estudiantes y convertirse en colaboradores activos del proceso investigativo.

Al Doctor Denis Alberto Castro Rodríguez, por su orientación y apoyo en el proceso de obtención de la beca y la inscripción a la Universidad de Antioquia.

A la profesora Luz Oneida Rentería Hilera, por su invaluable apoyo como codeudor en el proceso de la beca, sin lo cual no habría sido posible continuar con mis estudios de maestría.

A mis compañeros de la Maestría en Educación, línea Educación Matemática: Daniel Andrés Quiroz Vallejo, Jefferson Fabricio Palacios Rentería y Jhonatan Jiménez Gómez, por su apoyo incondicional y orientaciones valiosas que enriquecieron mi formación académica.

RESUMEN

Esta investigación caracterizó el Pensamiento Matemático manifestado por estudiantes de secundaria de la Escuela Normal Superior de Quibdó cuando resuelven tareas matemáticas mediante Python. Se adoptó un enfoque cualitativo con método fenomenológico, se trabajó con estudiantes de sexto, octavo y décimo grados. La recolección de datos incluyó análisis de códigos Python, entrevistas, grupos focales y observaciones de clase, y se abordaron los cinco tipos de pensamiento matemático propuestos por el MEN.

Los resultados revelan una caracterización multidimensional que evoluciona a través de cuatro dimensiones: representacional-simbólica, procedimental-estratégica, conceptual-estructural y metacognitiva-reflexiva. Se evidencia progresión desde pensamiento concreto-operacional en sexto grado hacia pensamiento relacional-sistémico en décimo. Se identificaron obstáculos epistemológicos relacionados con la traducción entre sistemas matemáticos tradicionales y computacionales.

Se concluye que Python funciona como mediador semiótico que transforma los procesos de pensamiento matemático, que facilita la representación de conceptos abstractos. Las implicaciones didácticas sugieren implementar estrategias diferenciadas por nivel educativo para desarrollar capacidades metacognitivas y optimizar la integración entre pensamiento matemático y computacional.

Palabras clave: Pensamiento Matemático, Python, educación matemática, pensamiento computacional, estudiantes de secundaria.

ABSTRACT

This research characterized the mathematical thinking exhibited by secondary school students from Escuela Normal Superior de Quibdó when solving mathematical tasks using Python. A qualitative approach, employing a phenomenological method, was adopted, working with students in sixth, eighth, and tenth grades. Data collection involved analyzing Python code, conducting interviews, focus groups, and classroom observations, all of which addressed the five types of mathematical thinking established by the Colombian Ministry of Education.

Results reveal a multidimensional characterization that evolves through four dimensions: representational-symbolic, procedural-strategic, conceptual-structural, and metacognitive-reflective—the findings evidence progression from concrete-operational thinking in sixth grade toward relational-systemic thinking in tenth grade. Epistemological obstacles were identified related to translation between traditional mathematical and computational systems.

It is concluded that Python functions as a semiotic mediator that transforms Mathematical Thinking processes, facilitating the externalization of abstract concepts. Didactic implications suggest implementing differentiated strategies by educational level and developing metacognitive capabilities to optimize the integration between mathematical and computational thinking.

Keywords: Mathematical Thinking, Python, mathematics education, characterization, computational thinking, secondary school students.

TABLA DE CONTENIDO

PRESENTACIÓN.....	12
CAPÍTULO 1.....	14
1. PLANTEAMIENTO DEL PROBLEMA	14
1.1. Caracterización del escenario educativo	16
1.2. Antecedentes	23
1.3. Formulación del problema	26
1.4. Justificación:	27
1.5. Objetivo General	29
1.6. Objetivos Específicos.....	29
CAPÍTULO 2.....	30
2. MARCO TEÓRICO.....	30
2.1. Fundamentos del pensamiento matemático	30
2.2. Caracterización del aprendizaje matemático	37
2.3. Tecnología en la Educación matemática.....	43
2.4. Python y pensamiento computacional	49
2.5. Herramientas y entornos de aprendizaje	52
2.6. Tendencias Emergentes	56
CAPÍTULO 3.....	59
3. DISEÑO METODOLÓGICO.....	59
3.1. Fundamentos Metodológicos de la Investigación.....	59
3.2. Diseño del Proceso de Investigación	59
3.3. Escenario de Investigación	61
3.4. Selección de Informantes Clave.....	62
3.5. Consideraciones Éticas	62

3.6.	Técnicas para la Recolección de la Información	64
3.7.	Uso de Inteligencia Artificial.....	65
3.8.	Evaluación del Pensamiento Matemático	66
3.9.	Análisis de Datos	66
3.10.	Caracterización del Pensamiento Matemático	69
3.11.	Consideraciones sobre la interpretación de procesos cognitivos.....	71
CAPÍTULO 4.....		73
4.	ANÁLISIS DE DATOS.....	73
4.1.	Consideraciones analíticas preliminares	73
4.2.	Análisis de las manifestaciones del pensamiento matemático en códigos Python ...	85
4.3.	Triangulación analítica profunda de los patrones identificados	117
4.4.	Caracterización integral del pensamiento matemático manifestado	126
4.5.	Implicaciones didácticas derivadas del análisis	135
4.6.	Visualización de patrones identificados.....	141
4.7.	Análisis de las entrevistas finales	143
CAPÍTULO 5.....		156
5.	CONCLUSIONES	156
5.1.	Cumplimiento de los objetivos de investigación	156
5.2.	Características principales del pensamiento matemático encontradas.....	157
5.3.	Cambios importantes según el nivel educativo.....	158
5.4.	Python como herramienta que cambia la forma de pensar matemáticas	159
5.5.	Experiencias y sentimientos de los estudiantes	160
5.6.	Dificultades y obstáculos encontrados.....	161
5.7.	Diferencias según el tipo de pensamiento matemático	162
5.8.	Implicaciones para la educación matemática.....	163

5.9.	Aportes al conocimiento sobre educación matemática	164
5.10.	Limitaciones del estudio	165
5.11.	Recomendaciones para futuras investigaciones.....	166
5.12.	Reflexiones finales.....	167
Anexos		169
Referencias bibliográficas.....		182

LISTA DE FIGURAS

Figura 1:	Uso de los resultados de evaluación de las Pruebas Saber	18
Figura 2:	Comparación de resultados Pruebas Saber grado 9 entre los años 2014-2017	19
Figura 3:	Resultados de Pruebas Saber grado Undécimo, para los años comprendidos entre 2019 al 2022.....	20
Figura 4:	Diseño de investigación	60
Figura 5:	Distribución de estudiantes participantes por género y grado académico	74
Figura 6:	Evolución del nivel de comprensión conceptual sobre programación en cuatro dimensiones a través de los tres grados analizados	76
Figura 7:	Distribución porcentual de percepciones sobre la utilidad de Python para el aprendizaje matemático según categorías y grados	78
Figura 8:	Distribución porcentual de intereses matemáticos específicos en estudiantes de sexto grado.....	80
Figura 9:	Distribución porcentual de intereses matemáticos específicos en estudiantes de octavo grado.....	80
Figura 10:	Distribución porcentual de intereses matemáticos específicos en estudiantes de décimo grado.....	81
Figura 11:	Tipos de experiencias iniciales con programación por grado	83
Figura 12:	Evolución de dificultades reportadas por los estudiantes según nivel educativo	85
Figura 13:		87
	Código de estudiante de sexto grado sobre horarios de lanchas en Quibdó	87
Figura 14:		
	Código sobre área y perímetro utilizado por un estudiante de sexto grado.	91
Figura 15:	95Código de conversión de unidades (kilogramos y gramos) usado por un estudiante de sexto grado... ..	95
Figura 16:	99Código de estudiante de octavo grado aplicando el Teorema de Pitágoras en el contexto de palafitos.....	99

Figura 17: Código utilizado por un estudiante de décimo grado para convertir series temporales (temperatura y precipitaciones).....	103
Figura 18: Código sobre modelación económica realizado por un estudiante de décimo grado (cálculo de costos pesqueros).....	108
Figura 19: Código de estudiante de sexto grado aplicando la fórmula del volumen cilíndrico.....	112
Figura 20: Representación en bloques (EduBlocks) del algoritmo para calcular el volumen del tanque, desarrollado por un estudiante de sexto grado.	116
Figura 21: Evolución del pensamiento matemático a través del código Python por nivel educativo.....	117
Figura 22: Comparativa de las cuatro dimensiones del pensamiento matemático por grado ..	135
Figura 23: Distribución de percepciones sobre dificultad en el uso de Python.....	146
Figura 24: Distribución porcentual de las categorías de aprendizajes matemáticos reportados por los estudiantes	148
Figura 25: Distribución de preferencias metodológicas.....	150
Figura 26: Correlación entre percepción de dificultad y preferencia metodológica	152

LISTA DE TABLAS

Tabla 1.	Características de los grupos participantes	73
Tabla 2.	Evolución conceptual sobre programación.....	75
Tabla 3.	Percepción de utilidad para el aprendizaje matemático	77
Tabla 4.	Intereses matemáticos por grado.....	79
Tabla 5.	Experiencias iniciales con programación.....	82
Tabla 6.	Evolución de dificultades reportadas	84
Tabla 7.	Categorización de aspectos de mayor interés	144
Tabla 8.	Percepción de dificultad en el uso de Python	145
Tabla 9.	Categorización de aprendizajes matemáticos reportados	147
Tabla 10.	Fuentes de confusión y estrategias de resolución	149
Tabla 11.	Preferencias metodológicas y sus justificaciones	150
Tabla 12.	Correlación entre dificultad percibida y preferencia metodológica.....	152

PRESENTACIÓN

Esta tesis de maestría fue realizada entre los años 2024 y 2025, en el programa Maestría en Educación, modalidad investigación, línea Educación Matemática, ofrecido por la Facultad de Educación de la Universidad de Antioquia, en la ciudad de Medellín.

En este documento se informa sobre las características del Pensamiento Matemático que manifiestan estudiantes de básica secundaria cuando resuelven tareas matemáticas mediante el lenguaje de programación Python. Los estudiantes de los grados sexto, octavo y décimo participaron de manera voluntaria, mediante el desarrollo de tareas contextualizadas y la realización de entrevistas, grupos focales y observaciones de clase.

Este trabajo presenta un diseño metodológico cualitativo con enfoque fenomenológico, en el cual interesa interpretar las experiencias vividas por los estudiantes al resolver problemas matemáticos usando Python, considerando las manifestaciones del pensamiento matemático, los procesos cognitivos subyacentes y los significados que los estudiantes atribuyen a los objetos matemáticos cuando se median por esta herramienta tecnológica. El documento consta de cinco capítulos, descritos a continuación.

El capítulo 1, Planteamiento del problema, presenta la problemática que define el interés por realizar la investigación, sustentada en referentes empíricos sobre los resultados de las pruebas estandarizadas en la Escuela Normal Superior de Quibdó, antecedentes teóricos sobre integración de programación en educación matemática, que confieren pertinencia e importancia a este trabajo. En este capítulo se definen la pregunta y los objetivos de la investigación.

El capítulo 2, Marco teórico, enuncia elementos teóricos y conceptuales relacionados con pensamiento matemático, tecnología educativa, Python como herramienta didáctica y pensamiento computacional. Estos elementos orientan la postura teórica frente a los conceptos relacionados con el problema de investigación.

El capítulo 3, Diseño metodológico, describe el enfoque fenomenológico, el paradigma interpretativo, las técnicas de recolección de información y los instrumentos empleados para orientar el proceso de indagación y cumplir los objetivos de investigación. También se presentan las características del escenario de investigación, la selección de informantes clave, las consideraciones éticas y las técnicas de análisis de datos utilizadas.

En el capítulo 4, Análisis de datos, se interpreta cualitativamente la información obtenida de los códigos Python desarrollados por los estudiantes, las entrevistas realizadas y las observaciones de clase. Se realiza la caracterización multidimensional del pensamiento matemático manifestado, la triangulación analítica de patrones identificados y se establecen las implicaciones didácticas derivadas del análisis.

En el capítulo 5, Conclusiones, se exponen los resultados obtenidos en el proceso investigativo, se da respuesta a la pregunta de investigación y se informa sobre el cumplimiento de los objetivos planteados. También se presentan las limitaciones del estudio, recomendaciones para futuras investigaciones y reflexiones finales sobre el potencial de Python en la educación matemática.

Adjunto a este trabajo se anexan los documentos: solicitud de autorización institucional, consentimientos informados, guías de reconocimiento de objetos y significados matemáticos y computacionales, tabla de características y objetos matemáticos, como información complementaria a los capítulos de la tesis.

CAPÍTULO 1

1. PLANTEAMIENTO DEL PROBLEMA

La sociedad se encuentra en transformación, caracterizada por avances tecnológicos y cambios en la forma en que procesamos y aplicamos el conocimiento. En este contexto, el sistema educativo enfrenta desafíos en la renovación de sus prácticas educativas, en el área de matemáticas, en el cual se evidencia una creciente brecha entre las metodologías tradicionales de enseñanza y las necesidades reales de los estudiantes en un mundo cada vez más digitalizado.

La educación matemática ofrece oportunidades para la innovación pedagógica y el desarrollo de habilidades de pensamiento. Según Gómez e Inagan (2014), este campo brinda a los docentes la oportunidad de transformar paradigmas de enseñanza, permitiéndoles usar herramientas y estrategias que fomenten el desarrollo de habilidades cognitivas. Esta visión se complementa con lo señalado por Cevallos y Cuadros (2015), quienes destacan que tanto profesores como estudiantes buscan en las matemáticas oportunidades para cultivar el sentido común, el espíritu crítico, la capacidad de discusión, la perseverancia y la cooperación en la búsqueda de soluciones.

La necesidad de transformar la enseñanza de las matemáticas se evidencia al considerar los resultados de investigaciones sobre el impacto de las habilidades matemáticas en el desarrollo profesional y económico de los individuos. Gholami y Muehlemann (2024) han mostrado que estas habilidades son necesarias para el éxito académico y para la movilidad social y el desarrollo profesional. Sus hallazgos indican que las competencias matemáticas están relacionadas con mejores oportunidades laborales y mayores ingresos a lo largo de la vida profesional.

En el contexto de la educación matemática, Ayora Carchi (2013) enfatiza la necesidad de desarrollar clases que incorporen recursos audiovisuales y estrategias didácticas para fomentar una

participación estudiantil activa. Esta necesidad se evidencia al considerar que los estudiantes son nativos digitales, acostumbrados a interactuar con tecnología en su vida cotidiana. La brecha entre los métodos de enseñanza y las expectativas de aprendizaje de los estudiantes puede resultar en desmotivación y bajo rendimiento académico.

La integración de tecnología, el uso de lenguajes de programación como Python, se presenta como una alternativa para abordar estos desafíos. Los estudios, como el realizado por Bustamante Narváez y Rodríguez Useche (2022), han mostrado resultados positivos en el uso de Python para fortalecer el pensamiento matemático de estudiantes de grado noveno. Sin embargo, es necesario examinar cómo esta integración podría influir en el desarrollo del pensamiento matemático en contextos como el de la Escuela Normal Superior de Quibdó.

Un problema identificado es la desconexión entre el aprendizaje matemático y su aplicación en contextos reales. Hernández et al. (2018) señalan que la educación, desde preescolar hasta los 11 años, debe enfocarse en desarrollar y fortalecer competencias, lo que requiere que los docentes rediseñen sus estrategias de enseñanza para adaptarse a estudiantes que tienden a desafiar situaciones problemáticas, hacer preguntas y presentar ideas críticas en función de sus intereses de aprendizaje.

La UNESCO (2021) refuerza esta perspectiva al reconocer el valor de la educación matemática para ampliar las oportunidades de los estudiantes, enfatizando que, si bien la acción y la reflexión individuales son importantes, es a través de las interacciones con otros que se facilita el aprendizaje de las matemáticas. Estas interacciones pueden incluir compañeros de clase, maestros, familiares, e incluso recursos tecnológicos como software educativo y herramientas de programación.

En el contexto colombiano, Maldonado (2010) señala que el desarrollo de competencias en educación básica y secundaria está fundamentado en la descripción, argumentación y proposición, según lo establecido por los entes rectores de la educación (MEN, ICFES). Este enfoque busca preparar a los estudiantes para el éxito académico y para su desempeño en escenarios de la vida real y su futuro profesional.

Según Reyes-Santander (2014), desarrollar el pensamiento matemático es un proceso complejo que involucra aspectos neuropsicológicos, y el estudiante debe desarrollar habilidades que van desde lo interno hasta cómo se relaciona con su entorno. Cuando se usan herramientas como Python en este proceso, los estudiantes aprenden matemáticas de manera diferente y adquieren habilidades de pensamiento computacional que están siendo demandadas en el ámbito laboral actual.

Esta situación se evidencia en instituciones educativas como la Escuela Normal Superior de Quibdó, donde los resultados de las pruebas estandarizadas sugieren la necesidad de implementar estrategias para mejorar el desarrollo del pensamiento matemático en los estudiantes. La integración de Python como herramienta didáctica representa una oportunidad para abordar esta necesidad de manera sistemática.

1.1. Caracterización del escenario educativo

La Escuela Normal Superior de Quibdó, fundada en 1936 y ubicada en el departamento del Chocó, municipio de Quibdó, constituye una institución educativa de importancia estratégica en la formación de docentes para la región del Pacífico colombiano. Durante sus más de ocho décadas

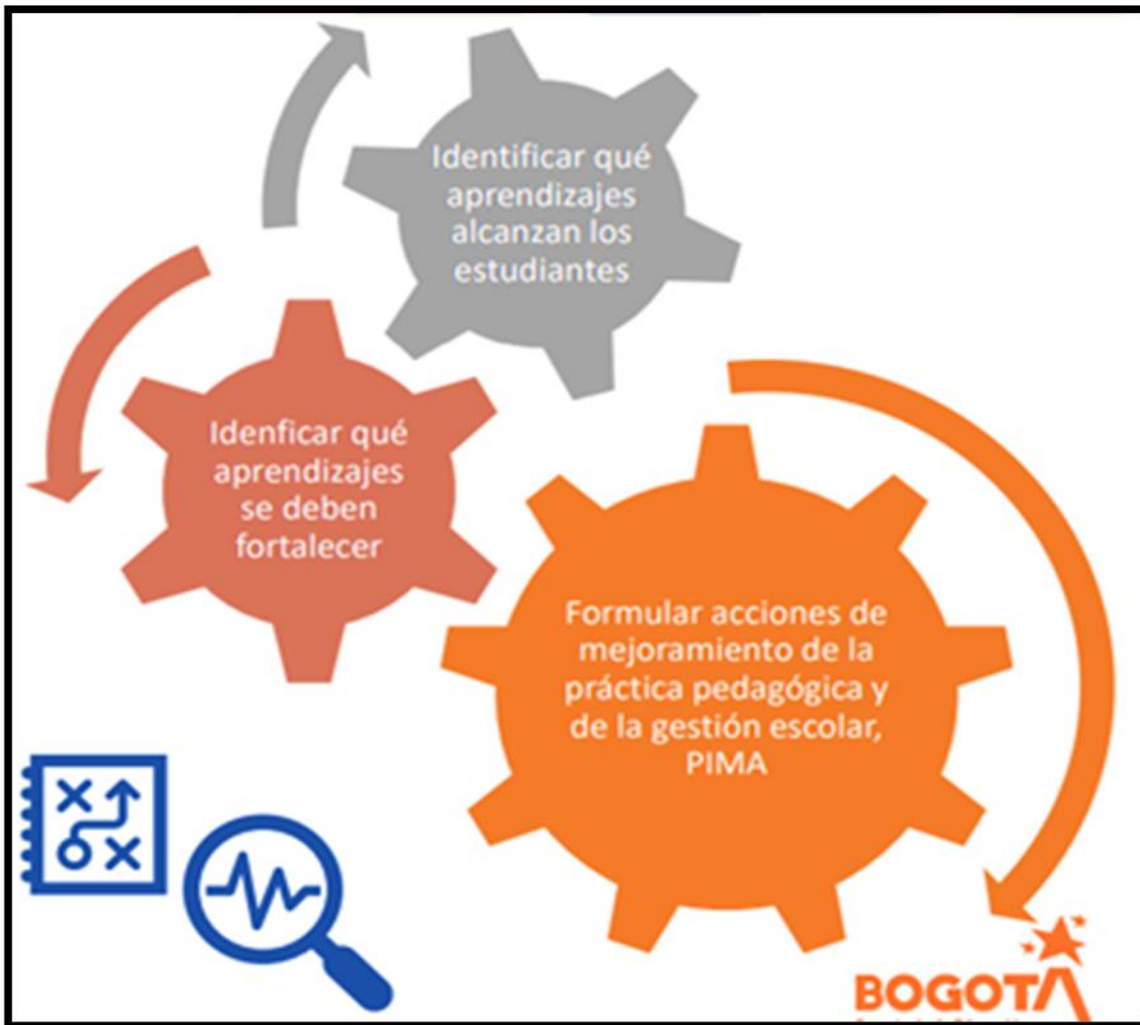
de funcionamiento, esta institución ha jugado un papel fundamental en el desarrollo educativo de la región, atendiendo a una población estudiantil diversa, predominantemente afrodescendiente e indígena. Su doble función como institución de educación media y centro de formación docente la posiciona de manera única para generar conocimiento que trascienda los límites de un estudio de caso particular y contribuya a la formación de educadores capaces de implementar innovaciones pedagógicas en territorios con características similares.

El contexto socioeconómico del Chocó presenta características particulares que configuran un escenario específico para los procesos educativos. Según datos del DANE (2023), el departamento mantiene uno de los índices más altos de necesidades básicas insatisfechas en Colombia, con una situación de pobreza que supera significativamente el promedio nacional de 12,9% en pobreza multidimensional. Adicionalmente, el Chocó registra la tasa más alta de pobreza monetaria del país con 67,7% en 2023 (DANE, 2024). Este contexto presenta desafíos que impactan directamente en el proceso educativo, afectando el acceso a recursos educativos y oportunidades de aprendizaje, pero también evidencia la necesidad de implementar estrategias educativas innovadoras para superar estas barreras socioeconómicas.

El Instituto Colombiano para la Evaluación de la Educación (ICFES) desarrolla una estrategia integral para la evaluación de la calidad educativa en Colombia. Como se observa en la Figura 1, el uso de los resultados de evaluación de las Pruebas Saber sigue un ciclo sistemático que incluye la identificación de fortalezas y oportunidades de mejora, el establecimiento de metas y acciones, la implementación de estrategias y el seguimiento al progreso. Este proceso permite a las instituciones educativas tomar decisiones informadas para mejorar sus prácticas pedagógicas y resultados académicos.

Figura 1:

Uso de los resultados de evaluación de las Pruebas Saber

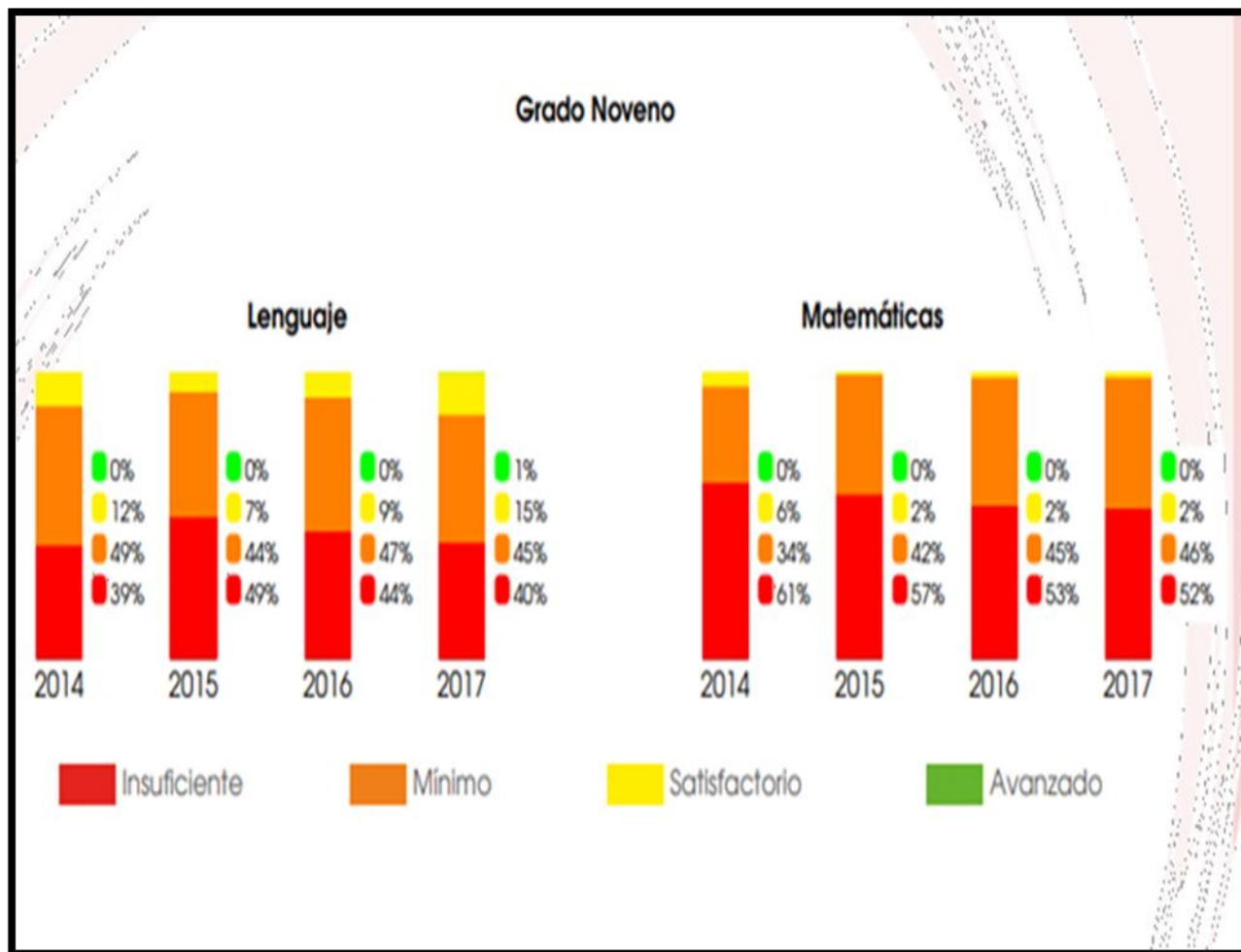


Nota: Se muestra el ciclo sistemático de evaluación implementado por el ICFES. Fuente: ICFES (2021).

Los resultados en matemáticas para el grado noveno entre 2014 y 2017 han mostrado una concentración persistente de estudiantes en los niveles más básicos de desempeño, como se evidencia en la Figura 2, con más del 50% ubicándose en el nivel Insuficiente (variando entre 52% y 61%), entre 34% y 46% en el nivel Mínimo, mientras que solo entre 2% y 6% logra un nivel Satisfactorio, y 0% alcanza el nivel Avanzado durante todo el período analizado.

Figura 2:

Comparación de resultados Pruebas Saber grado 9 entre los años 2014-2017



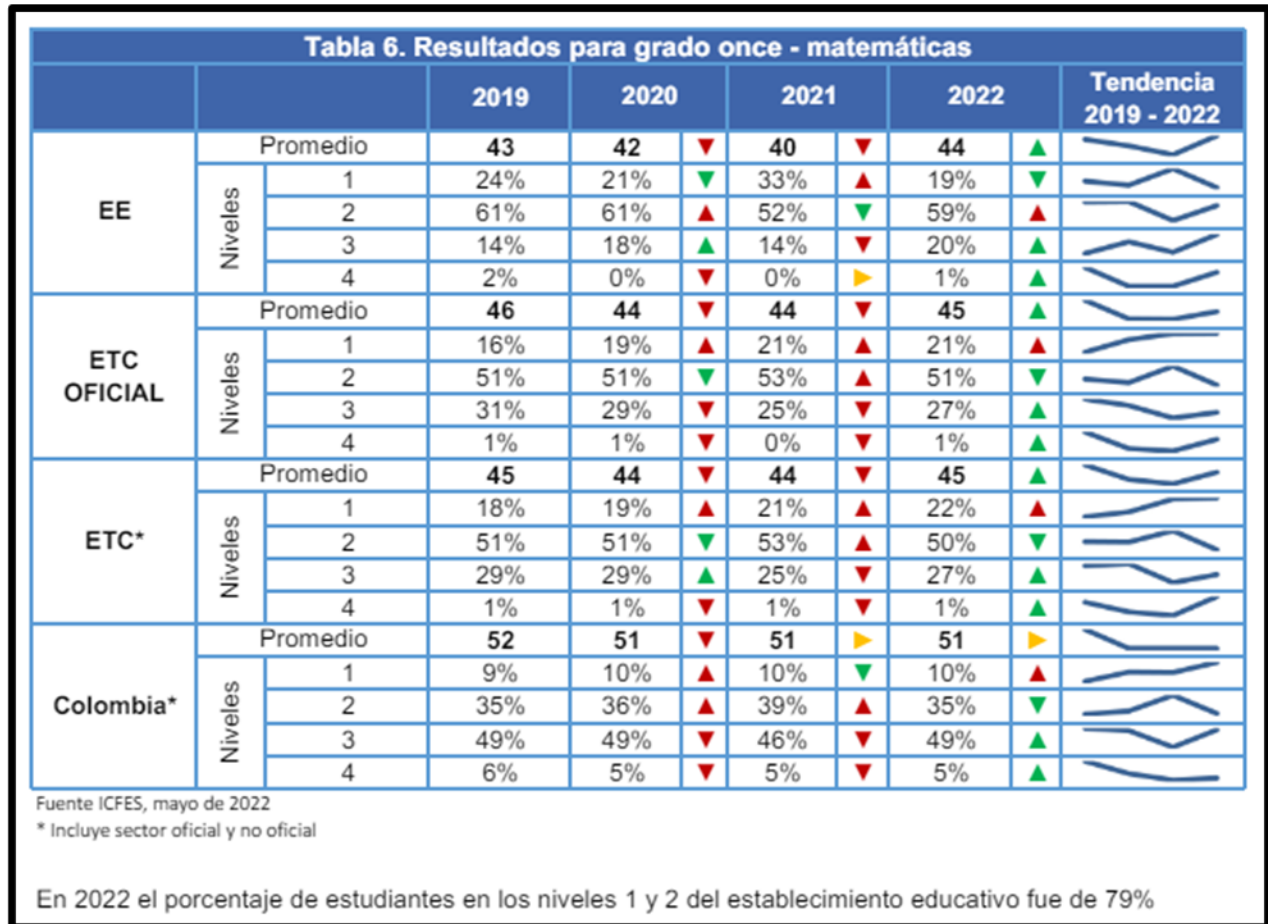
Nota: La gráfica muestra la distribución porcentual de estudiantes por niveles de desempeño. Fuente: http://superate20.edu.co/resultados_saber/ DANE de la Institución: 127001000446

Esta situación se mantiene en los resultados de undécimo grado (2019-2022), en el cual se observa que el nivel más bajo (Nivel 1) presenta fluctuaciones, pasando de 24% en 2019 a 19% en 2022, mientras que el Nivel 2 se mantiene consistentemente alto, variando entre 52% y 61%. Los niveles superiores muestran un comportamiento limitado, con el Nivel 3 oscilando entre 14% y 20%, y el Nivel 4 manteniéndose en niveles mínimos, reduciéndose de 2% en 2019 a 1% en 2022.

En conjunto, los niveles más bajos (1 y 2) representan consistentemente alrededor del 78-85% de los estudiantes durante todo el período analizado.

Figura 3:

Resultados de Pruebas Saber grado Undécimo, para los años comprendidos entre 2019 al 2022



Nota: La figura presenta la evolución temporal del desempeño en matemáticas para el grado undécimo.
Fuente: https://diae.mineducacion.gov.co/dia_e/index.php DANE de la Institución: 127001000446

Sin embargo, es crucial interpretar estos resultados no como indicadores definitivos de un proceso educativo deficiente, sino como evidencia de la necesidad de explorar estrategias pedagógicas innovadoras que respondan a las particularidades del contexto educativo chocoano. Las evaluaciones estandarizadas, aunque proporcionan información valiosa sobre competencias

específicas, tienen limitaciones para capturar la riqueza del pensamiento matemático que se desarrolla en contextos culturales específicos. El análisis de los resultados revela una concentración persistente de estudiantes en los niveles más básicos de desempeño, lo que sugiere la necesidad de fortalecer el desarrollo de competencias matemáticas fundamentales.

En este sentido, los resultados de la institución abren una oportunidad para implementar y evaluar enfoques pedagógicos alternativos que puedan potenciar el desarrollo del pensamiento matemático desde perspectivas más contextualizadas e innovadoras. El panorama educativo contemporáneo ha sido transformado por la integración de tecnologías digitales, en el área de matemáticas. Según Borba et al. (2016), "si los medios cambian, el conocimiento también puede cambiar", estableciendo una relación directa entre las herramientas tecnológicas disponibles y las formas de construir comprensión matemática. La evidencia internacional sugiere que la integración de herramientas computacionales en la educación matemática puede facilitar nuevas formas de representación, exploración y comprensión de conceptos abstractos, promoviendo un aprendizaje más activo y contextualizado. St Omer et al. (2025) señalan que las herramientas digitales "permiten a los estudiantes visualizar y explorar conceptos matemáticos... haciendo las ideas abstractas más accesibles" (p. 2) y que "estas tecnologías no solo ayudan en la comprensión conceptual, sino que también fomentan el aprendizaje activo y la indagación" (p. 2). Además, destacan que los dispositivos móviles se han utilizado para "contextualizar conceptos complejos mediante actividades al aire libre, donde los estudiantes toman fotografías de objetos para investigar área, perímetro, simetría y ángulos" (p. 2). Esta perspectiva es respaldada por el meta-análisis de Sokolowski et al. (2015), quienes encontraron que "las tecnologías permiten computaciones complejas y modelado dinámico que conducen a formas más experimentales de enseñanza y aprendizaje de las matemáticas" (p. 2), facilitando que los estudiantes "aprecien

aplicaciones de herramientas matemáticas en situaciones de la vida real" (p. 2) mediante exploraciones que "involucran al aprendiz con el entorno a través de acciones definidas de recopilación e investigación de información" (pp. 2-3), con un tamaño del efecto de 0.60 (p. 1).

En este contexto de búsqueda de alternativas pedagógicas innovadoras, la programación emerge como una oportunidad para el desarrollo del pensamiento matemático. Los lenguajes de programación como Python ofrecen posibilidades únicas para que los estudiantes visualicen conceptos abstractos, experimenten con patrones matemáticos y desarrollen habilidades de resolución de problemas de manera interactiva. Las investigaciones previas han demostrado que el uso de herramientas de programación puede contribuir significativamente al desarrollo de habilidades de razonamiento lógico, pensamiento algorítmico y resolución de problemas matemáticos.

Sin embargo, una revisión de la literatura revela la ausencia de investigaciones que aborden el uso de programación en el contexto específico del municipio de Quibdó y el departamento del Chocó. Este vacío investigativo es preocupante considerando que la región cuenta con instituciones educativas estratégicas como las Escuelas Normales, que tienen el potencial de multiplicar cualquier innovación pedagógica a través de los futuros docentes que forman. Mientras que existe una amplia literatura internacional sobre programación en educación matemática, y algunos estudios emergentes en contextos urbanos colombianos, los territorios del Pacífico colombiano permanecen inexplorados desde esta perspectiva investigativa. Esta ausencia de estudios deja un vacío considerable en el conocimiento sobre cómo estas herramientas podrían funcionar específicamente en contextos afrodescendientes e indígenas del Pacífico colombiano, privando a la comunidad académica y educativa de evidencia empírica para la toma de decisiones pedagógicas informadas en estos territorios.

La escasez de investigación sobre programación en educación matemática en el contexto chocoano representa tanto un desafío como una oportunidad única. Por un lado, la ausencia de referentes locales específicos genera incertidumbre sobre la efectividad y pertinencia de estas herramientas en este contexto particular. Por otro lado, esta misma escasez configura una ventana de oportunidad para contribuir de manera pionera al conocimiento sobre la intersección entre pensamiento matemático, programación y contextos culturales específicos del Pacífico colombiano.

En este escenario, la exploración del pensamiento matemático mediado por herramientas de programación como Python en la Escuela Normal Superior de Quibdó representa una oportunidad para llenar un vacío investigativo significativo y generar evidencia empírica que informe el diseño de estrategias pedagógicas pertinentes y contextualizadas para la región.

1.2. Antecedentes

La caracterización institucional y contextual revela dos elementos clave: los desafíos particulares del contexto educativo chocoano y la escasez de investigaciones sobre programación en educación matemática en territorios del Pacífico colombiano. Esta situación hace necesaria una revisión de la literatura que identifique marcos teóricos y experiencias investigativas para informar el diseño de intervenciones pedagógicas contextualizadas.

En este sentido, las investigaciones recientes han mostrado resultados prometedores en la integración de la programación, específicamente Python, en la enseñanza de las matemáticas. Bustamante Narváez y Rodríguez Useche (2022) desarrollaron un estudio con estudiantes de grado noveno, en el cual implementaron Python como herramienta didáctica fundamentada en la Teoría de las Situaciones Didácticas. Sus resultados mostraron mejoras significativas en los niveles de institucionalización y validación de los saberes matemáticos, a pesar de las dificultades iniciales

que enfrentaron los estudiantes. Esta investigación destaca cómo la incorporación de estrategias de enseñanza apoyadas en las TIC contribuye al desarrollo del pensamiento crítico y matemático.

Por su parte, Rodríguez Vanegas y Orjuela Díaz (2022) estudiaron la integración del lenguaje de programación Python en la Educación Básica Secundaria en Colombia, encontrando que constituye una estrategia innovadora y efectiva para promover el desarrollo del pensamiento variacional y computacional. Sus hallazgos indican que el uso de Python mejora la comprensión de objetos matemáticos y también genera mayor motivación en el aprendizaje de las matemáticas.

La importancia de desarrollar competencias matemáticas sólidas se hace evidente en el estudio de Gholami y Muehlemann (2024), quienes encontraron que las habilidades matemáticas son un factor determinante en la selección de individuos en empresas de formación y en el desarrollo salarial posterior. Su investigación, realizada en el contexto del sistema dual de formación profesional alemán, demuestra que las habilidades matemáticas son cruciales para la clasificación en empresas de formación de alta calidad, y generan un rendimiento salarial sustancial en los primeros años como trabajadores calificados.

En el contexto de la educación primaria, Mendoza Aguirre (2018) realizó un estudio sobre el impacto del software de programación en el desarrollo del pensamiento lógico matemático. Sus resultados revelaron mejoras significativas en diversas áreas: un aumento del 17% en el desarrollo del pensamiento lógico matemático, 13% en la competencia en situaciones de cantidad, 16% en regularidad, equivalencia y cambio, 18% en forma, movimiento y localización, y 21% en gestión de datos e incertidumbre. Aunque este estudio se realizó en primaria, sus hallazgos sugieren el potencial de la programación como herramienta para desarrollar habilidades matemáticas en niveles educativos superiores.

Arotuma y Soraya (2017) han señalado que la enseñanza de la programación contribuye significativamente a que los estudiantes desarrollen capacidades para resolver problemas complejos y situaciones de la vida real. Sus investigaciones destacan cómo la programación ofrece oportunidades únicas para desarrollar el pensamiento y conocimientos matemáticos clave, además de proporcionar herramientas que ayudan a desarrollar el pensamiento creativo y lógico-matemático.

Briz Redón y Serrano Aroca (2018) observaron que la implementación de contenidos algebraicos a través de la programación promovió la revisión y profundización de conceptos matemáticos. Su estudio demostró que las clases donde se introdujo a los estudiantes en la programación sirvieron simultáneamente para enseñar y repasar numerosos contenidos matemáticos, creando así un ambiente de aprendizaje integrado y significativo.

Maggio (2018) enfatiza la importancia de desarrollar competencias para el siglo XXI en la educación matemática. Su investigación destaca cómo la integración de herramientas tecnológicas mejora el aprendizaje de las matemáticas, por otro lado, desarrollan habilidades críticas como el pensamiento crítico, la creatividad, la resolución de problemas y el pensamiento computacional, competencias en la era digital.

Supelano Mesa (2023) realizó un estudio significativo sobre la implementación de Python en el aula de matemáticas, demostrando cómo esta herramienta facilita la transición de procesos aritméticos a algebraicos. Sus hallazgos revelan que Python proporciona un puente efectivo entre el pensamiento concreto y el abstracto, permitiendo a los estudiantes visualizar y comprender mejor los conceptos algebraicos.

Pertejo-López (2017) investigó el impacto de la programación en el desarrollo del pensamiento lógico-matemático, proporcionando evidencia empírica sobre cómo las herramientas

de programación pueden potenciar el aprendizaje de las matemáticas. Su estudio destaca cómo la programación ayuda a los estudiantes a desarrollar una comprensión más profunda de los conceptos matemáticos a través de la experimentación activa y la resolución de problemas prácticos.

1.3. Formulación del problema

El análisis de la caracterización institucional y contextual ha revelado la necesidad de explorar estrategias pedagógicas innovadoras que puedan potenciar el desarrollo del pensamiento matemático en la Escuela Normal Superior de Quibdó. Esta institución, ubicada en un contexto socioeconómico caracterizado por desafíos particulares del Pacífico colombiano y una población estudiantil predominantemente afrodescendiente e indígena, presenta una oportunidad única para implementar enfoques educativos contextualizados que respondan a las particularidades territoriales y culturales de la región.

La revisión de antecedentes investigativos sugiere que la integración de Python en la enseñanza de las matemáticas puede ofrecer una alternativa para abordar estas dificultades. Diversos estudios han demostrado el potencial de este lenguaje de programación para promover el desarrollo del pensamiento matemático, facilitando la visualización de conceptos abstractos, la experimentación con patrones matemáticos y el desarrollo de habilidades de resolución de problemas de manera interactiva.

Sin embargo, estas investigaciones se han realizado principalmente en contextos diferentes al de la Escuela Normal Superior de Quibdó. La institución presenta características particulares en términos de su población estudiantil predominantemente afrodescendiente e indígena, los recursos disponibles y los desafíos socioeconómicos específicos del Pacífico colombiano, factores que pueden influir significativamente en cómo los estudiantes interactúan con esta herramienta

tecnológica y desarrollan su pensamiento matemático. Además, la ausencia de estudios sobre el uso de programación en educación matemática en el contexto chocoano constituye un vacío investigativo que requiere ser abordado.

En este sentido, antes de implementar cualquier intervención pedagógica, resulta fundamental comprender cómo los estudiantes de este contexto específico desarrollan su pensamiento matemático al utilizar Python como herramienta mediadora. Esta comprensión permitirá no solo caracterizar las formas particulares en que se manifiesta el pensamiento matemático en este contexto, sino también generar conocimiento que informe el diseño de intervenciones pedagógicas más efectivas y pertinentes para esta población estudiantil.

Por lo tanto, surge la necesidad de caracterizar el pensamiento matemático de estos estudiantes cuando utilizan Python, considerando sus particularidades contextuales y necesidades específicas. En este contexto, se plantea la siguiente pregunta de investigación:

¿Cuáles son algunas características del pensamiento matemático manifestado por estudiantes de básica secundaria de la Escuela Normal Superior de Quibdó cuando resuelven tareas matemáticas mediante el programa Python?

1.4. Justificación:

Este estudio encuentra pertinencia en la necesidad de incorporar el lenguaje de programación Python para comprender el Pensamiento Matemático manifestado por estudiantes de básica secundaria cuando resuelven tareas matemáticas en la Escuela Normal Superior de Quibdó. Esta incorporación brinda a los estudiantes oportunidades para resolver tareas matemáticas, fomenta la motivación y desarrolla competencias para el siglo XXI, como el pensamiento crítico, la creatividad, la resolución de problemas y el pensamiento computacional (Maggio, 2018).

La justificación del estudio radica en que el uso de Python como herramienta didáctica en el aula de matemáticas representa una alternativa para fortalecer la resolución de problemas y favorecer la transición de procesos aritméticos a algebraicos (Supelano Mesa, 2023). Además, la programación en Python contribuye al desarrollo de competencias como el pensamiento lógico, la formalización algebraica y la capacidad de abordar situaciones problemáticas en contextos reales (Bustamante Narváez & Rodríguez Useche, 2022).

En una perspectiva teórica, el estudio se sustenta en los planteamientos de autores como Reyes-Santander (2014), quien enfatiza que el pensamiento matemático es una capacidad construida, resultado de procesos neuropsicológicos que requieren de habilidades tanto internas como externas del individuo. Asimismo, investigaciones (Pertejo-López, 2017; Bustamante Narváez & Rodríguez Useche, 2022; Rodríguez Vanegas & Orjuela Díaz, 2022) han mostrado que el uso de la programación, Python, tiene el potencial de ofrecer oportunidades para desarrollar pensamientos y conocimientos matemáticos.

En síntesis, esta investigación se justifica por su relevancia, al abordar una problemática educativa y proponer una alternativa para comprender el desarrollo del Pensamiento Matemático de los estudiantes de secundaria cuando resuelven tareas matemáticas mediante el lenguaje de programación Python, en el contexto de la Escuela Normal Superior de Quibdó. Los resultados de este estudio contribuirán a la generación de conocimiento sobre el uso de Python como herramienta didáctica en la enseñanza de las matemáticas, y servirán de base para el diseño de estrategias que promuevan un aprendizaje contextualizado.

1.5. Objetivo General

Caracterizar el Pensamiento Matemático manifestado por estudiantes de básica secundaria de la Escuela Normal Superior de Quibdó cuando resuelven tareas matemáticas mediante el programa Python.

1.6. Objetivos Específicos

- ❖ Diseñar tareas matemáticas específicas para ser resueltas mediante Python, que permitan conocer características del Pensamiento Matemático de los estudiantes.
- ❖ Analizar configuraciones de objetos y significados matemáticos manifestados por los estudiantes cuando resuelven tareas matemáticas mediante el uso de Python, identificando características específicas del Pensamiento Matemático.
- ❖ Describir conflictos de significado asociados al Pensamiento Matemático de los estudiantes al trabajar con tareas matemáticas en Python.

CAPÍTULO 2

2. MARCO TEÓRICO

En este capítulo se precisa el significado de los conceptos clave que se usan en esta investigación, donde se destaca la noción del pensamiento matemático, sus características y manifestaciones, las tareas matemáticas mediadas por tecnología, y el uso de Python como herramienta didáctica. También se presentan las clasificaciones y elementos de la literatura sobre la caracterización del pensamiento matemático en entornos de programación, con atención a la integración de tecnologías educativas en la enseñanza de las matemáticas.

2.1. Fundamentos del pensamiento matemático

2.1.1. Noción de pensamiento matemático

El pensamiento matemático es un concepto complejo que ha sido objeto de estudio desde múltiples perspectivas teóricas. Según Reyes-Santander (2014), constituye un "producto final de variados procesos neuropsicológicos, estos procesos provienen y se desarrollan en diferentes áreas y requieren de diferentes habilidades del individuo en su interior y del individuo en el medio" (p. 145). Esta conceptualización enfatiza la naturaleza multidimensional del pensamiento matemático y su desarrollo en el individuo.

Schoenfeld (1992) amplía esta perspectiva al señalar que el pensamiento matemático trasciende la resolución de problemas, abarcando capacidades como el razonamiento, la argumentación, la comunicación y la representación de ideas matemáticas. Su enfoque integra tanto procesos cognitivos, como la toma de decisiones y la planificación, como metacognitivos, incluyendo la autorregulación y la reflexión sobre el propio aprendizaje.

Mason, Burton y Stacey (2010) complementan esta visión al caracterizar el pensamiento matemático como un proceso dinámico que involucra la exploración, la conjetura, la justificación

y la generalización. Estos autores enfatizan que constituye un proceso de construcción de significado a través de la interacción con problemas y situaciones matemáticas, más que la aplicación de algoritmos.

Esta perspectiva se alinea con lo planteado por Flores Samaniego (2017), quien describe el pensamiento matemático como una forma del pensamiento reflexivo que se utiliza al hacer matemáticas, incorporando razonamientos inductivos, deductivos y abductivos que adquieren características según el contexto matemático. Además, señala que este tipo de pensamiento dota al individuo de habilidades de razonamiento abstracto que permiten resolver problemas, utilizando las herramientas tecnológicas disponibles.

En síntesis, el pensamiento matemático emerge como un constructo complejo que integra procesos cognitivos y metacognitivos, habilidades de razonamiento, capacidades de representación y comunicación, todo ello enmarcado en un proceso dinámico de construcción de significados matemáticos. Este pensamiento se desarrolla y manifiesta tanto en las estructuras mentales del individuo como en su interacción con el entorno y las herramientas matemáticas disponibles.

2.1.2. Características y Componentes del Pensamiento Matemático

El pensamiento matemático "implica un producto de la integración de distintas modalidades sensoriales, cognitivas y de representaciones. Además, el desarrollo del Pensamiento Matemático se puede ver reflejado en la estructura y en la forma, que van tomando los conocimientos, en el individuo" (Reyes-Santander, 2014, p. 145).

Los conceptos centrales en los modelos cognitivos del Pensamiento Matemático son:

- La memoria y los procesos de información
- El conocimiento y las representaciones

- Las estructuras cognitivas y representaciones mentales

El Pensamiento Matemático utiliza un lenguaje y metodología propios, y su estructura se expresa en las formas del lenguaje y las representaciones que genera (Reyes-Santander, 2014).

Profundizando en la estructura del pensamiento matemático, es importante considerar los componentes específicos que permiten su caracterización y evaluación en contextos educativos.

2.1.3. Componentes del Pensamiento Matemático

Para caracterizar el pensamiento matemático de los estudiantes, Kilpatrick, Swafford y Findell (2001) identifican cinco componentes que permiten una comprensión de cómo se desarrolla y manifiesta este tipo de pensamiento:

Conceptualización: La capacidad de comprender y manipular conceptos matemáticos abstractos, como funciones, ecuaciones y propiedades geométricas. En el contexto de Python, se analizará cómo los estudiantes utilizan el lenguaje para representar y manipular estos conceptos.

Fluidez procedimental: La habilidad para aplicar procedimientos y algoritmos matemáticos. Con Python, se explorará cómo los estudiantes implementan algoritmos para resolver problemas matemáticos.

Estrategias de Resolución de Problemas: La capacidad de planificar y ejecutar estrategias para resolver problemas matemáticos. Se analizará cómo los estudiantes utilizan Python para descomponer problemas complejos en partes más manejables y cómo aplican diferentes estrategias para llegar a soluciones.

Razonamiento y Pruebas: La habilidad para formular argumentos matemáticos y justificar las soluciones. Con Python, se explorará cómo los estudiantes utilizan el lenguaje para verificar sus soluciones y cómo justifican sus procedimientos.

Comunicación Matemática: La capacidad de expresar ideas matemáticas de manera clara y precisa. En este estudio, se analizará cómo los estudiantes utilizan Python para comunicar sus soluciones, ya sea a través de código, gráficos o explicaciones verbales.

Estos componentes se manifiestan de diversas formas según el tipo de pensamiento matemático que se esté desarrollando. El Ministerio de Educación Nacional (1998) establece una clasificación de cinco tipos de pensamiento matemático que permiten organizar y estructurar estas manifestaciones.

2.1.4. Tipos de Pensamiento Matemático

De acuerdo con los Lineamientos Curriculares de Matemáticas del Ministerio de Educación Nacional (1998), el pensamiento matemático se subdivide en cinco tipos:

Pensamiento numérico: Refiere a la comprensión del uso y significados de los números y la numeración, el sentido y significado de las operaciones, y el desarrollo de estrategias de cálculo y estimación. Este tipo de pensamiento permite:

- Representar la numerosidad de las magnitudes a través de diversos medios y sistemas
- Razonar con y sobre los números para identificar, argumentar y justificar relaciones
- Comunicar ideas sobre números, sus relaciones y operaciones
- Formular y ejercitar procedimientos para operar y relacionar cantidades

Pensamiento espacial: Considerado como "el conjunto de los procesos cognitivos mediante los cuales se construyen y se manipulan las representaciones mentales de los objetos del espacio, las relaciones entre ellos, sus transformaciones, y sus diversas traducciones o representaciones materiales" (MEN, 1998, p. 56). Incluye:

- La representación de lugares, trayectorias y objetos bidimensionales y tridimensionales
- El razonamiento sobre localización, trayectorias y formas geométricas
- La comunicación de ideas espaciales en diversos lenguajes
- La proposición y ejercitación de procedimientos de construcción y medición

Pensamiento métrico: Se refiere a la comprensión general de las magnitudes, su medición y el uso flexible de los sistemas métricos o de medidas en diferentes situaciones (MEN, 1998).

Este pensamiento permite:

- Representar cantidades y medidas en actividades de la vida diaria
- Razonar sobre conceptos de estimación y magnitud en situaciones cotidianas
- Comunicar ideas sobre conceptos de cantidad y medición
- Proponer estrategias y procedimientos para comparar tamaños y establecer relaciones

Pensamiento variacional: Involucra el reconocimiento, percepción, identificación y caracterización de la variación y el cambio en diferentes contextos, así como su descripción, modelación y representación en distintos sistemas o registros simbólicos (MEN, 1998). Este pensamiento permite:

- Representar procesos de variación y cambio usando diferentes sistemas
- Razonar sobre cambio, variación, covariación y sistemas de relaciones
- Comunicar ideas sobre cambio y variación mediante diversas representaciones
- Proponer y analizar procedimientos para modelar fenómenos de variación

Pensamiento aleatorio: Ayuda a tomar decisiones en situaciones de incertidumbre, azar y riesgo, utilizando conceptos y procedimientos de la teoría de probabilidades y estadística inferencial (MEN, 1998). Este pensamiento permite:

- Razonar en situaciones de incertidumbre
- Conectar conceptos estadísticos
- Combinar ideas de probabilidad y azar
- Comunicar ideas usando lenguaje estadístico y probabilístico

2.1.5. Habilidades del Pensamiento Matemático

El desarrollo de habilidades básicas del pensamiento propicia en los alumnos un aprendizaje más perdurable, significativo y de mayor aplicabilidad en la toma de decisiones y en la solución de problemas relacionados con situaciones de la vida diaria (Zerpa, 2003). Estas habilidades son relevantes en la educación matemática, donde el pensamiento sistemático y estructurado juega un papel.

Sánchez (1995) identifica las siguientes habilidades:

Observación: Es un proceso mental que implica la identificación de características de los estímulos y su integración en una imagen mental coherente. Requiere:

- Definición clara del propósito de observación
- Identificación y enumeración sistemática de características
- Verificación rigurosa de los resultados obtenidos

Descripción: Consiste en la capacidad de transmitir ordenadamente los datos o características obtenidos tras la observación. El proceso incluye:

- Organización de características según criterios específicos
- Formulación clara y sistemática de la descripción

- Verificación de la completitud y precisión de lo descrito

Comparación y Relación: Son procesos complementarios que permiten establecer conexiones entre diferentes elementos matemáticos. Involucran:

- Identificación de variables relevantes
- Establecimiento de diferencias y semejanzas significativas
- Elaboración de conexiones lógicas entre elementos

Clasificación: Proceso que permite ordenar elementos según criterios determinados, siguiendo una secuencia:

- Definición de propósitos clasificatorios
- Identificación de características
- Agrupación sistemática por clases o categorías
- Verificación de la coherencia clasificatoria

Análisis y Síntesis: Habilidades complementarias que permiten:

- Descomponer un todo en sus partes constituyentes
- Comprender relaciones entre componentes
- Integrar elementos en un todo coherente
- Establecer patrones y regularidades

Planteamiento y Verificación de Hipótesis: Capacidades que incluyen:

- Formulación de conjeturas matemáticas
- Diseño de estrategias de verificación
- Validación o refutación de propuestas
- Generalización de resultados

Definición de Conceptos: Habilidad que permite:

- Identificar características
- Establecer relaciones jerárquicas
- Formular definiciones precisas
- Validar la coherencia conceptual

Estas habilidades no operan de manera aislada, sino que se interrelacionan y complementan entre sí, formando un sistema integrado de pensamiento matemático. Como señala Du Boulay (1980), el desarrollo de estas habilidades tiene un papel especial en estudiantes con dificultades matemáticas, pues el trabajo sistemático con ellas proporciona una experiencia estructurada y convincente que motiva el aprendizaje.

La implementación efectiva de estas habilidades requiere una planificación cuidadosa por parte del docente y un ambiente de aprendizaje que promueva la exploración y el razonamiento matemático. Como indica González (2001), el conocimiento lógico matemático se construye al relacionar las experiencias obtenidas en la manipulación de objetos y la generación de hipótesis, procesos que se facilitan cuando se desarrollan sistemáticamente estas habilidades básicas del pensamiento.

2.2. Caracterización del aprendizaje matemático

2.2.1. Perspectivas Didácticas

La conceptualización de las matemáticas propuesta por Kaput & Roschelle (2013) ofrece una visión integral que trasciende la mera manipulación de números y fórmulas. Los autores definen las matemáticas como "un estudio culturalmente compartido de patrones y lenguajes que se aplican y extienden mediante formas sistemáticas de razonamiento y argumentación". Esta definición enfatiza la naturaleza social y cultural del conocimiento matemático, así como su carácter sistemático y argumentativo.

En su análisis, los autores identifican cinco aspectos que estructuran el pensamiento matemático. El primero es la medición y cuantificación, que abarca la comprensión de magnitudes y unidades, el desarrollo de sistemas de medición, la interpretación de cantidades en contexto y las relaciones entre diferentes medidas.

El segundo aspecto es el cambio cuantificable y variación, que incluye el análisis de patrones de cambio, el estudio de relaciones funcionales, la comprensión de tasas y proporciones, y la modelación de fenómenos dinámicos.

La incertidumbre especificable constituye el tercer aspecto, englobando el análisis probabilístico, la gestión de datos estadísticos, la evaluación de riesgos y la toma de decisiones bajo incertidumbre.

El cuarto aspecto se refiere al espacio y características espaciales, que comprende el entendimiento de formas y figuras, las relaciones geométricas, la visualización espacial y las transformaciones geométricas.

Finalmente, los algoritmos y estructuras abstractas conforman el quinto aspecto, incluyendo el desarrollo de procedimientos sistemáticos, la construcción de modelos matemáticos, el análisis de patrones abstractos y la manipulación de estructuras formales.

2.2.2. Caracterización de Tareas Matemáticas

Las tareas matemáticas constituyen un elemento estructural en el proceso de enseñanza-aprendizaje, que constituye un medio para el desarrollo del pensamiento matemático. Según Yeo (2017), la calidad y naturaleza del aprendizaje matemático de los estudiantes está ligada con las características y demandas cognitivas de las tareas que se les presentan. Partiendo de esta premisa, el autor desarrolla un marco analítico comprehensivo que permite caracterizar y analizar la apertura de las tareas matemáticas a través de cinco dimensiones

La Respuesta de la Tarea: Esta dimensión contempla dos categorías principales:

Tareas Cerradas: Se caracterizan por tener respuestas determinadas y claramente identificables. En estas tareas, existe un conjunto específico y limitado de respuestas correctas que los estudiantes deben alcanzar. Este tipo de tareas resulta útil para evaluar la comprensión de conceptos específicos y el dominio de procedimientos matemáticos básicos.

Tareas Abiertas: Pueden ser de dos tipos:

Bien definidas: Presentan múltiples respuestas posibles, pero todas ellas son objetivamente evaluables según criterios establecidos.

Mal definidas: Permiten respuestas subjetivas que pueden variar según la interpretación y el enfoque del estudiante.

El Objetivo de la Tarea: Esta dimensión se refiere a la meta o propósito que se busca alcanzar:

a) Objetivos Cerrados: Presentan una meta concreta y claramente definida. Los estudiantes tienen una comprensión clara de lo que deben lograr y los criterios para determinar si han alcanzado el objetivo.

b) Objetivos Abiertos: Se subdividen en:

Bien definidos: Presentan múltiples metas específicas, todas ellas válidas y alcanzables.

Mal definidos: Tienen objetivos más vagos o generales, permitiendo diferentes interpretaciones sobre lo que constituye una resolución exitosa.

El Método de Solución: Esta dimensión aborda las estrategias y procedimientos que los estudiantes pueden emplear:

a) Método Cerrado: Implica un procedimiento rutinario y predeterminado. Los estudiantes siguen una secuencia específica de pasos para llegar a la solución.

- b) Método Abierto Bien Definido:** Permite múltiples aproximaciones y métodos para llegar a la misma respuesta. Cada método es válido siempre que conduzca a una solución correcta.
- c) Método Abierto Mal Definido:** No existe un método específico predeterminado, y los estudiantes deben desarrollar sus propias estrategias de resolución.

La Complejidad: Esta dimensión considera el nivel de dificultad y la estructura de la tarea:

- a) Complejidad Cerrada:** Son tareas simples con una estructura clara y directa. Los estudiantes pueden abordarlas con conocimientos y habilidades básicas.
- b) Complejidad Abierta Dependiente del Sujeto:** La dificultad puede reducirse mediante andamiaje apropiado. El nivel de complejidad varía según el apoyo y la orientación proporcionados al estudiante.
- c) Complejidad Abierta Inherente:** Son tareas que mantienen un alto nivel de complejidad incluso con apoyo estructural. Requieren un pensamiento matemático avanzado y habilidades de resolución de problemas sofisticadas.

2.2.3. Niveles Cognitivos en Tareas Matemáticas

Stein y Smith (2009) proponen una clasificación de las tareas matemáticas basada en sus demandas cognitivas, estableciendo una distinción clara entre tareas de bajo y alto nivel. Esta categorización proporciona un marco para comprender la profundidad y complejidad del pensamiento matemático requerido en diferentes actividades.

Las Tareas de Bajo Nivel comprenden dos categorías principales que se centran en aspectos más básicos del aprendizaje matemático. Las tareas de memorización involucran la reproducción de hechos, reglas y definiciones, sin requerir una comprensión conceptual profunda, centrándose en la retención y recuperación de información. Por otro lado, los procedimientos sin conexión

implican la aplicación de algoritmos estandarizados, donde el énfasis está en obtener respuestas correctas más que en la comprensión conceptual, con poca conexión a los conceptos matemáticos subyacentes.

En contraste, las Tareas de Alto Nivel exigen un compromiso cognitivo más profundo. Los procedimientos con conexiones requieren una comprensión conceptual significativa, estableciendo vínculos entre diferentes representaciones matemáticas y demandando explicaciones y justificaciones que promueven el pensamiento matemático significativo. La categoría más avanzada, "haciendo matemáticas", implica un pensamiento matemático complejo y no algorítmico, requiriendo exploración y comprensión profunda de conceptos matemáticos, junto con la autorregulación del proceso de pensamiento.

2.2.4. Dimensiones y evaluación del aprendizaje matemático

El análisis y evaluación del aprendizaje matemático requiere un marco estructurado que permita comprender sus diferentes dimensiones. Ponte (2005) establece un marco que se articula a través de dos dimensiones que permiten categorizar y analizar las experiencias de aprendizaje matemático.

La primera dimensión se centra en el Grado de Desafío, que se manifiesta en dos niveles distintos. El desafío reducido comprende tareas que requieren un nivel básico de compromiso cognitivo, mientras que el desafío elevado implica tareas que demandan un pensamiento matemático más sofisticado y complejo. Esta gradación permite adaptar las actividades al nivel de desarrollo de los estudiantes mientras se mantiene un equilibrio apropiado en la demanda cognitiva.

La segunda dimensión aborda la Estructura de las tareas matemáticas. Las tareas cerradas presentan un camino claro y definido hacia la solución, con parámetros y objetivos bien

establecidos. En contraste, las tareas abiertas ofrecen múltiples vías de exploración y resolución, fomentando la creatividad y el pensamiento divergente en los estudiantes.

Complementando este marco dimensional, Ponte y Quaresma (2012) profundizan en la importancia del contexto en el aprendizaje matemático, identificando tres tipos. El contexto de realidad vincula las matemáticas con situaciones auténticas del mundo real; el contexto de semi-realidad presenta situaciones simuladas pero verosímiles; y el contexto matemático se centra en las relaciones y estructuras puramente matemáticas. Esta clasificación contextual enriquece la comprensión de cómo los estudiantes interactúan y aplican los conceptos matemáticos en diferentes situaciones.

2.2.5. Factores del Aprendizaje Matemático

La comprensión de los factores que influyen en el aprendizaje matemático permite desarrollar estrategias educativas. Un elemento determinante es la motivación, aspecto destacado por Hidalgo (2018), quien observa que "los estudiantes aprueban la asignatura solo porque se sienten obligados y no por estar motivados a aprender, lo cual evidencia la importancia de la motivación como factor para favorecer el aprendizaje" (p. 131). Esta realidad subraya la necesidad de transformar la percepción de las matemáticas de una obligación académica a una herramienta práctica.

La relevancia del pensamiento matemático trasciende el ámbito académico. Como señala Suárez (2019), su desarrollo se evidencia en la vida cotidiana, en la resolución de problemas reales. Esta conexión entre el aprendizaje matemático y su aplicación práctica favorece el desarrollo de competencias duraderas.

La integración tecnológica emerge como otro factor determinante en el proceso de aprendizaje. Santos-Trigo & Camacho Machín (2013) enfatizan que "es ampliamente reconocido

que el uso de la tecnología computacional ofrece a profesores y estudiantes diferentes formas de representar y explorar problemas o conceptos matemáticos" (p. 279). Esta diversificación de las formas de representación y exploración:

- Facilita la comprensión de conceptos abstractos
- Permite múltiples aproximaciones a un mismo problema
- Proporciona herramientas para la visualización y manipulación de conceptos
- Enriquece las estrategias de enseñanza y aprendizaje
- Promueve la exploración activa y el descubrimiento

2.3. Tecnología en la Educación matemática

2.3.1. Tecnología Educativa y Matemáticas

La educación en una sociedad caracterizada por un desarrollo tecnológico avanzado experimenta el impacto de las Tecnologías de la Información y la Comunicación (TIC). Como señala Cabero (2001), incorporar las TIC a la educación es una necesidad, donde la discusión debe orientarse a determinar estrategias para ofrecer mejores oportunidades de educación matemática e integrarlas de manera que los objetivos educativos trasciendan los tecnológicos.

García-González et al. (2020) indican que la integración de las TIC en el proceso de enseñanza-aprendizaje de la Matemática Escolar puede contribuir a:

- Mejorar el rendimiento académico
- Desarrollar el pensamiento matemático
- Adquirir habilidades en los estudiantes

2.3.2. Evolución de la Tecnología en Educación Matemática

La evolución de la tecnología en la educación matemática ha experimentado una transformación significativa a lo largo de las últimas décadas. Borba et al. (2016) han identificado

y caracterizado cuatro fases distintivas que marcan el desarrollo progresivo de la integración tecnológica en la enseñanza de las matemáticas:

Primera Fase: Introducción de Logo como Herramienta, esta fase inicial marcó un hito en la integración de la tecnología en la educación matemática con la introducción del lenguaje de programación Logo. Este período se caracterizó por:

- La primera aproximación sistemática a la programación educativa
- El desarrollo del pensamiento procesal en matemáticas
- La introducción de conceptos geométricos a través de comandos simples
- El establecimiento de las bases para el pensamiento computacional en matemáticas

Segunda Fase: Software especializado la siguiente etapa se distinguió por el surgimiento de software específicamente diseñado para la educación matemática, como Cabri y Geometer's Sketchpad. Esta fase se caracterizó por:

- El desarrollo de interfaces gráficas intuitivas
- La manipulación directa de objetos matemáticos
- La exploración dinámica de conceptos geométricos
- La visualización interactiva de propiedades matemáticas

Tercera Fase: Surgimiento de internet La llegada de Internet transformó las posibilidades de la educación matemática, introduciendo:

- Acceso a recursos matemáticos en línea
- Comunicación y colaboración a distancia
- Nuevas formas de representación matemática
- Expansión de las fuentes de información y aprendizaje

Cuarta Fase: Web 2.0, Internet Banda Ancha y Aprendizaje Móvil. La fase actual se caracteriza por una integración más profunda y ubicua de la tecnología, que incluye:

- Plataformas interactivas de aprendizaje
- Aplicaciones matemáticas móviles
- Recursos educativos abiertos
- Aprendizaje colaborativo en línea
- Herramientas de visualización y simulación avanzadas

2.3.3. Integración Tecnológica

Pierce, Stacey, Wander & Ball (2011) proporcionan un análisis comprehensivo sobre cómo las herramientas digitales potencian diversos aspectos del aprendizaje matemático. Su investigación identifica cuatro áreas donde la tecnología transforma la experiencia educativa.

En primer lugar, la investigación de objetos matemáticos se ve enriquecida por las capacidades de visualización y manipulación que ofrecen las herramientas digitales. Los estudiantes pueden explorar propiedades, relaciones y comportamientos matemáticos de manera dinámica e interactiva, permitiendo una comprensión más profunda de conceptos abstractos.

La resolución de problemas alcanza nuevas dimensiones con el uso de herramientas sofisticadas, que permiten a los estudiantes abordar problemas más complejos y realistas. Estas herramientas facilitan los cálculos, y permiten explorar múltiples estrategias de solución y visualizar los resultados de manera inmediata.

El desarrollo de la comprensión conceptual se potencia a través de las múltiples representaciones que la tecnología hace posibles. Los estudiantes pueden ver y manipular conceptos matemáticos desde diferentes perspectivas, estableciendo conexiones más sólidas entre representaciones abstractas y concretas.

Finalmente, el razonamiento en la exploración de problemas matemáticos se fortalece mediante la capacidad de las herramientas digitales para facilitar la experimentación, la formulación de conjeturas y la verificación de hipótesis. Los estudiantes pueden desarrollar y probar sus ideas matemáticas de manera sistemática y eficiente.

2.3.4. Impacto en la Comprensión Matemática

La mediación tecnológica ha transformado la naturaleza de la comprensión matemática en el entorno educativo contemporáneo. Como señalan Goos & Geiger (2012, citado en Borba et al., 2016), esta mediación genera "formas cualitativamente diferentes de conocimiento y relaciones entre estudiantes, profesores y la propia tecnología" (p.25). Esta transformación se manifiesta en múltiples niveles de la experiencia educativa, redefiniendo cómo se aprende, qué se aprende y cómo se construyen las relaciones pedagógicas.

Este cambio se ha vuelto relevante en el contexto actual, donde, como afirman Dainamang et al. (2024), "en la era digital, el uso de la tecnología se ha vuelto inevitable en todos los aspectos de la vida, incluido el mundo de la educación" (p. 111). Esta inevitabilidad ha dado lugar a:

- Nuevas formas de visualización y representación matemática
- Transformación en las interacciones educativas
- Cambios en los roles tradicionales de docentes y estudiantes
- Desarrollo de nuevas estrategias de resolución de problemas
- Emergencia de formas innovadoras de evaluación y retroalimentación

La integración de la tecnología en la comprensión matemática ha modificado las herramientas disponibles, además ha redefinido cómo se construye y se accede al conocimiento matemático en el contexto educativo moderno.

2.3.5. Retos en la Implementación

La implementación de tecnología en la educación matemática enfrenta desafíos que van más allá de la simple integración de herramientas digitales. Hegedus & Moreno-Armella (2009) presentan una perspectiva crítica y transformadora al argumentar que "la tecnología está aquí para transformar el pensamiento, y no para servir como un dispositivo protésico que sostenga viejos estilos de pedagogía o estándares curriculares" (p. 398). Esta afirmación desafía la tendencia a utilizar la tecnología solo como una extensión de métodos tradicionales.

Este planteamiento implica una reconceptualización de la práctica educativa, donde la tecnología debe catalizar una transformación profunda en la forma en que se desarrolla el pensamiento matemático, más que limitarse a replicar o automatizar prácticas existentes. El verdadero reto radica en aprovechar el potencial transformador de la tecnología para crear nuevos paradigmas de enseñanza y aprendizaje.

La implementación efectiva requiere un cambio de mentalidad tanto en docentes como en instituciones educativas, abandonando la visión de la tecnología como una herramienta auxiliar y adoptándola como un medio para revolucionar la forma en que se construye y comprende el conocimiento matemático. Este proceso implica repensar las metodologías de enseñanza, y además los objetivos educativos y las formas de evaluación del aprendizaje.

2.3.6. Aprendizaje en Contextos Digitales

La transformación digital del aprendizaje matemático representa un cambio paradigmático en la educación. Como señalan Borba et al. (2016), existe una relación entre los medios y la naturaleza del conocimiento, estableciendo que "si los medios cambian, el conocimiento también puede cambiar". Esta perspectiva se materializa en el aprendizaje matemático en entornos

digitales, que se construye sobre la base de la práctica social e interacción, donde "el aprendizaje está situado en la práctica y toda práctica es esencialmente social por naturaleza" (p.17).

Los entornos de aprendizaje abiertos han emergido como espacios dinámicos que transforman la experiencia educativa tradicional. Estos entornos digitales facilitan múltiples dimensiones del aprendizaje matemático, comenzando por la participación en actividades creativas. Esta dimensión incluye el desarrollo de proyectos matemáticos innovadores, la exploración de conceptos a través de simulaciones interactivas, la creación de representaciones matemáticas personalizadas y la experimentación con diferentes aproximaciones a la resolución de problemas.

El desarrollo de redes de aprendizaje constituye otra dimensión, que abarca la construcción de comunidades de práctica matemática, el intercambio de conocimientos entre pares, la colaboración en proyectos matemáticos y el acceso a recursos y expertos en diferentes áreas.

La organización de contenidos matemáticos emerge como una tercera dimensión crucial, que comprende la estructuración personalizada del material de aprendizaje, la creación de repositorios de conocimiento, la vinculación de conceptos relacionados y el desarrollo de mapas conceptuales interactivos.

Finalmente, la gestión de interacciones educativas completa estas dimensiones, incluyendo la coordinación de actividades colaborativas, la facilitación de discusiones matemáticas, la implementación de sistemas de retroalimentación y el seguimiento del progreso individual y grupal.

2.4. Python y pensamiento computacional

2.4.1. Python como Herramienta Didáctica

Python es un lenguaje de programación multi-paradigma que se ha perfilado como una opción recomendada tanto para la enseñanza de la programación como para el desarrollo de software en general. De acuerdo con Challenger-Pérez et al. (2014), "Python es un lenguaje de alto nivel ya que contiene implícitas algunas estructuras de datos como listas, diccionarios, conjuntos y tuplas, que permiten realizar algunas tareas complejas en pocas líneas de código y de manera legible" (p. 3).

La sintaxis de Python es destacada por su sencillez en comparación con otros lenguajes, lo cual tiene gran importancia en el ámbito educativo (Challenger-Pérez et al., 2014). Entre sus características principales se encuentran un intérprete interactivo, una extensa librería estándar, extensibilidad en C/C++ y licencia de código abierto, siendo ampliamente utilizado en la programación universitaria, la industria y la comunidad científica.

2.4.2. Impacto de Python en la Educación Matemática

Según Altamirano et al. (2023), "Python cuenta con una amplia gama de bibliotecas matemáticas, como NumPy, SciPy, SymPy y Matplotlib, que facilitan la realización de cálculos matemáticos y la visualización de resultados. Estas bibliotecas proporcionan herramientas para álgebra lineal, optimización, estadísticas, cálculo simbólico, graficación y más" (p. 367).

Briz Redón y Serrano Aroca (2018) señalan que la implementación de contenido algebraico a través de la programación "obligó a la revisión de cuestiones de divisibilidad, optimización y funciones" (p. 159). La enseñanza de la programación contribuye a que los estudiantes tengan oportunidades de resolver problemas complejos y solucionar problemas de la vida real (Arotuma y Soraya, 2017).

2.4.3. Relación entre Pensamiento Computacional y Matemático

Weintrop et al. (2016) establecen un marco teórico que identifica cuatro prácticas que vinculan el pensamiento computacional con el pensamiento matemático. Esta interrelación demuestra cómo ambas formas de pensamiento se complementan y refuerzan mutuamente en el proceso de aprendizaje.

La primera práctica, el análisis de datos, representa la capacidad de interpretar, manipular y extraer significado de conjuntos de información. En este contexto, las habilidades matemáticas se combinan con las computacionales para desarrollar métodos sistemáticos de análisis y comprensión de patrones en los datos.

El modelado y simulación constituye la segunda práctica, donde los estudiantes aprenden a crear representaciones abstractas de sistemas y fenómenos. Esta práctica integra el pensamiento matemático para formular modelos y el pensamiento computacional para implementarlos y experimentar con ellos.

La resolución de problemas computacionales emerge como la tercera práctica, combinando estrategias algorítmicas con razonamiento matemático. Esta práctica implica la capacidad de descomponer problemas complejos en componentes manejables y diseñar soluciones sistemáticas.

Finalmente, el pensamiento sistémico representa la capacidad de comprender y trabajar con sistemas complejos. Esta práctica integra conceptos matemáticos con principios computacionales para analizar las interrelaciones y comportamientos de sistemas complejos.

2.4.4. Programación y Pensamiento Matemático

La relación entre programación y pensamiento matemático ha sido objeto de análisis profundo en el campo educativo. Du Boulay (1980) presenta una perspectiva dual sobre el valor

de la programación interactiva en el desarrollo del pensamiento matemático, identificando dos dimensiones que transforman la manera en que los estudiantes interactúan con las matemáticas.

La primera dimensión se centra en el contenido matemático, donde las matemáticas se convierten en el objeto central de estudio. En este enfoque, la programación sirve como herramienta para explorar conceptos matemáticos, permitiendo a los estudiantes analizar estructuras matemáticas a través del código. El contenido matemático se hace explícito y manipulable, y la abstracción matemática se materializa en estructuras programables que los estudiantes pueden examinar y modificar directamente.

La segunda dimensión se enfoca en la actividad matemática, o lo que Du Boulay (1980) denomina 'matematización'. Este aspecto enfatiza la programación como medio para hacer matemáticas activamente, facilitando el desarrollo de procesos de pensamiento matemático y la implementación práctica de conceptos teóricos. Los estudiantes participan en la experimentación y verificación de hipótesis matemáticas, construyendo activamente su conocimiento a través de la interacción con el código y los conceptos matemáticos.

2.4.5. Procesos Matemáticos según Du Boulay(1980)

Los procesos matemáticos, según el análisis de Du Boulay(1980), se pueden categorizar en dos grupos que reflejan diferentes aspectos del pensamiento matemático. Esta categorización brinda un marco estructurado para comprender cómo los estudiantes interactúan y desarrollan conceptos matemáticos.

Los procesos de contenido comprenden tres elementos: Adición, Transformación e Integración. La Adición implica la incorporación de nuevos elementos al conocimiento existente, la Transformación involucra la modificación y adaptación de conceptos matemáticos, y la

Integración se refiere a la síntesis de diferentes elementos matemáticos en un todo coherente. Estos procesos trabajan en conjunto para construir una comprensión profunda y estructurada del contenido matemático.

Por otro lado, los procesos de actividad se manifiestan a través de la Resolución de problemas, la Generación de pruebas y la Búsqueda de patrones. Estos procesos representan las habilidades activas que los estudiantes desarrollan y aplican en su trabajo matemático. La resolución de problemas implica el desarrollo de estrategias y métodos de solución, la generación de pruebas involucra la construcción de argumentos matemáticos rigurosos, y la búsqueda de patrones fomenta el desarrollo del pensamiento inductivo y la generalización matemática.

2.5. Herramientas y entornos de aprendizaje

2.5.1. ‘Google Colaboratory’ como Entorno de Aprendizaje

‘Google Colaboratory’, comúnmente conocido como ‘Colab’, representa una plataforma educativa que ha transformado la manera en que se enseña y aprende programación. Según da Silva (2024), es "un entorno digital disponible en la nube, gratuito y alojado por Google" (p. 3), que se ha convertido en una herramienta para la enseñanza de la programación, especialmente en el contexto educativo matemático.

Esta plataforma se distingue por su capacidad de integrar múltiples aspectos del proceso de aprendizaje en un solo entorno. A través de su sistema de ‘notebooks’, los estudiantes pueden desarrollar código, documentar su proceso de pensamiento, experimentar con diferentes soluciones y visualizar resultados, todo en un espacio de trabajo unificado. Los notebooks de Colaboratory combinan la potencia del código Python con la capacidad de incluir texto explicativo, imágenes y visualizaciones, creando así un documento interactivo y dinámico.

2.5.2. Características Técnicas

‘Google Colaboratory’ presenta un conjunto de características técnicas avanzadas que lo convierten en una plataforma robusta y versátil para la enseñanza y el aprendizaje de la programación. Estas características técnicas pueden clasificarse en varios aspectos.

En primer lugar, la plataforma implementa una división modular de bloques de texto y código a través de una estructura modular sofisticada. Esta característica permite una separación clara entre contenido explicativo y código ejecutable, una organización jerárquica del contenido, la capacidad de reordenar y reestructurar bloques según necesidades, la ejecución selectiva de segmentos de código y una documentación integrada en formato ‘Markdown’.

El segundo aspecto es el acceso a GPU gratuito para computación. El soporte computacional incluye procesamiento acelerado por ‘hardware’ para tareas intensivas, asignación dinámica de recursos computacionales, capacidad de procesamiento paralelo, optimización automática de recursos y soporte para operaciones matemáticas complejas.

La capacidad para escribir código Python en navegador web constituye otro aspecto crucial. La implementación basada en web ofrece un editor de código con resaltado de sintaxis, autocompletado inteligente de código, depuración en tiempo real, gestión de errores interactiva y acceso a todas las funcionalidades de Python sin necesidad de instalación local.

Finalmente, la plataforma se destaca por su integración con servicios de ‘Machine Learning’, proporcionando bibliotecas preinstaladas de aprendizaje automático, herramientas de procesamiento de datos, ‘frameworks’ de ‘deep learning’, capacidades de visualización avanzada e integración con APIs de inteligencia artificial.

2.5.3. Ventajas Educativas Según Dainamang et al. (2024), Colaboratory:

Las ventajas educativas de Google Colaboratory han sido extensamente analizadas por Dainamang et al. (2024), quienes identifican múltiples beneficios pedagógicos que hacen de esta plataforma una herramienta valiosa en el contexto educativo.

En primer lugar, la plataforma destaca por su capacidad de personalización del aprendizaje. Como señalan los autores, Colaboratory "permite a los estudiantes estudiar material de manera independiente, a un ritmo y de una manera que les resulte más adecuada" (p. 112). Esta característica de aprendizaje autodirigido que:

- Permite a los estudiantes avanzar según su propio nivel de comprensión
- Facilita la revisión y práctica de conceptos según necesidades individuales
- Promueve la autonomía y responsabilidad en el proceso de aprendizaje
- Respeta los diferentes estilos y velocidades de aprendizaje

En segundo lugar, Colaboratory demuestra ser una herramienta efectiva para la introducción a la programación. Los investigadores destacan que la plataforma "puede introducir a los estudiantes a conceptos de programación y ser una herramienta útil para ayudar en el proceso de aprendizaje" (p. 112). Este aspecto es relevante porque:

- Proporciona un entorno amigable para principiantes
- Permite una transición gradual hacia conceptos más complejos
- Integra teoría y práctica de manera natural
- Ofrece retroalimentación inmediata durante el proceso de aprendizaje

Finalmente, uno de los aspectos más significativos es su capacidad para facilitar el aprendizaje colaborativo. La plataforma promueve el "diálogo, negociación, debate para resolver sus problemas" (p. 113). Esta dimensión colaborativa se manifiesta en:

- La posibilidad de trabajar en proyectos grupales en tiempo real
- El intercambio de ideas y soluciones entre estudiantes
- La construcción colectiva del conocimiento
- El desarrollo de habilidades de comunicación y trabajo en equipo

Estas ventajas educativas se complementan entre sí, creando un ecosistema de aprendizaje que combina la flexibilidad del estudio independiente con las ricas oportunidades de colaboración y el desarrollo estructurado de habilidades de programación.

2.5.4. EduBlocks como entorno de programación visual

EduBlocks es una plataforma educativa de código abierto desarrollada por Joshua Lowe, diseñada para facilitar la transición entre la programación por bloques visuales y la codificación textual en Python. Esta herramienta permite a los estudiantes construir programas mediante el ensamblaje de bloques gráficos que se convierten automáticamente en código Python válido.

El uso de entornos de programación visual como EduBlocks se fundamenta en la necesidad de reducir la carga cognitiva asociada con la sintaxis, permitiendo a los estudiantes centrarse en los aspectos conceptuales de la resolución de problemas. Como señalan Weintrop y Wilensky (2017), los entornos basados en bloques pueden facilitar la comprensión inicial de conceptos computacionales antes de la transición a la programación textual.

En el contexto de esta investigación, EduBlocks se utilizará principalmente con los estudiantes de sexto grado, proporcionando un andamiaje visual que les permitirá abordar tareas matemáticas más complejas, como el cálculo de volúmenes cilíndricos. La plataforma actuará como un puente entre la manipulación concreta y la abstracción formal, permitiendo a los estudiantes visualizar las estructuras de código mientras desarrollan su pensamiento matemático.

La implementación de EduBlocks responderá a las limitaciones tecnológicas identificadas en la Escuela Normal Superior de Quibdó, ofreciendo una alternativa accesible que funciona en navegadores web con requisitos técnicos mínimos. Esta característica resultará especialmente relevante dado el contexto de recursos tecnológicos limitados identificado en la caracterización institucional de la investigación.

2.6. Tendencias Emergentes

2.6.1. Inteligencia artificial y educación matemática

La integración de la inteligencia artificial (IA) en la educación matemática representa una evolución significativa en los métodos de enseñanza y aprendizaje. Como señala Rouhiainen (2018), la IA se define como "la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano" (p. 17). Esta definición establece las bases para comprender el potencial transformador de la IA en el contexto educativo.

El alcance de la IA en educación es amplio y multifacético. Como observa Varela Uribe (2021), citando a Russell et al. (2010), "la IA abarca una gran variedad de subcampos, que van desde lo general, aprendizaje y percepción, hasta lo específico" (p. 8). Esta diversidad de aplicaciones permite abordar diferentes aspectos del proceso educativo, desde la personalización del aprendizaje hasta la evaluación automatizada de competencias matemáticas.

En el contexto específico de la educación, Padilla (2019) identifica tres enfoques de aplicación que están transformando la enseñanza de las matemáticas. El primer enfoque involucra los agentes de software conversacionales inteligentes o chatbots, que proporcionan apoyo personalizado y respuestas inmediatas a las consultas de los estudiantes. El segundo enfoque se centra en las plataformas online para auto-aprendizaje, que permiten a los estudiantes avanzar a su

propio ritmo con contenido adaptativo. Finalmente, la robótica educativa ofrece experiencias prácticas y tangibles que conectan los conceptos matemáticos con aplicaciones del mundo real.

2.6.2. Desarrollo de Competencias

Hoyles, Noss, Kent & Bakker (2010) presentan un análisis sobre cómo los avances tecnológicos están redefiniendo los entornos profesionales y, consecuentemente, transformando las exigencias en los sistemas educativos. Su investigación destaca la necesidad de desarrollar competencias integradas que respondan a estas nuevas demandas.

Las capacidades de comunicación emergen como una competencia en el contexto tecnológico-matemático. Los estudiantes deben desarrollar habilidades para expresar ideas matemáticas utilizando diversos medios y herramientas digitales, facilitando la colaboración y el intercambio de conocimientos. Esta comunicación se vuelve crucial en entornos profesionales cada vez más interconectados.

Las conexiones entre usuarios representan otro aspecto de las competencias modernas. La tecnología facilita la creación de redes de aprendizaje y colaboración, donde los estudiantes pueden compartir conocimientos, resolver problemas conjuntamente y construir comprensión matemática de manera colectiva.

La integración de simulaciones en el aprendizaje matemático permite a los estudiantes explorar conceptos complejos de manera interactiva. Esta competencia incluye la capacidad de crear, manipular y analizar modelos matemáticos dinámicos, proporcionando una comprensión más profunda de los fenómenos estudiados.

Las diversas representaciones de conceptos matemáticos constituyen una competencia crucial en el entorno tecnológico actual. Los estudiantes deben desarrollar la capacidad de trabajar

con múltiples representaciones, comprendiendo cómo diferentes visualizaciones y formatos pueden revelar distintos aspectos de un mismo concepto matemático.

CAPÍTULO 3

3. DISEÑO METODOLÓGICO

3.1. Fundamentos Metodológicos de la Investigación

El presente estudio se enmarca en el paradigma interpretativo y sigue un enfoque cualitativo, basado en la exploración de las experiencias vividas por los estudiantes al resolver tareas matemáticas mediante el lenguaje de programación Python. Este enfoque permite comprender las percepciones y vivencias subjetivas de los participantes, lo cual permite caracterizar el pensamiento matemático en un contexto específico (Hernández Sampieri, 2014).

El método fenomenológico fue seleccionado para esta investigación, ya que permite explorar, describir y comprender las experiencias vividas por los estudiantes con respecto al fenómeno del pensamiento matemático manifestado al resolver tareas con Python. Según Hernández Sampieri (2014), la fenomenología busca revelar la esencia de las experiencias desde la perspectiva de los participantes, lo cual es crucial para entender cómo los estudiantes interactúan con la herramienta tecnológica y cómo esta influye en su desarrollo cognitivo.

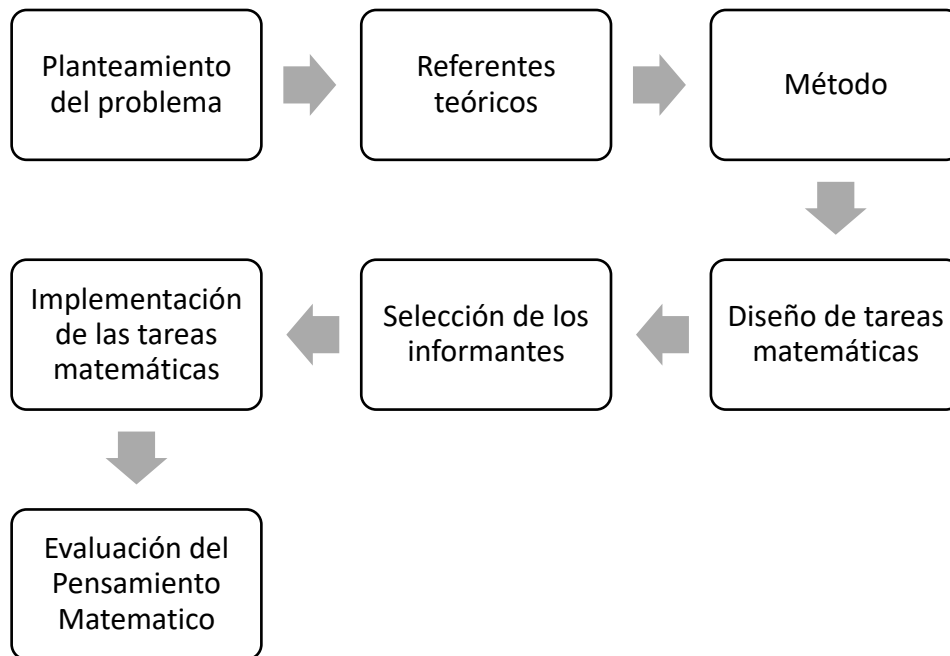
3.2. Diseño del Proceso de Investigación

El diseño de investigación se establece en función de la pregunta central: *¿Cuáles son algunas características del Pensamiento Matemático manifestado por estudiantes de básica secundaria de la Escuela Normal Superior de Quibdó cuando resuelven tareas matemáticas mediante el programa Python?* Siguiendo a Salgado (2007), en el enfoque cualitativo, el diseño es flexible y abierto, ajustándose a las condiciones del escenario y a la evolución de los acontecimientos durante la investigación.

El diseño se estructura en las siguientes fases:

Figura 4:

Diseño de investigación



Nota: El diagrama representa la secuencia metodológica estructurada que guía el desarrollo de la investigación, desde el planteamiento del problema hasta la evaluación del pensamiento matemático.

El diseño de la investigación sigue una secuencia metodológica estructurada que parte del planteamiento del problema, avanza por los referentes teóricos y el método, para luego concretarse en acciones específicas, como se muestra en la Figura 4.

La ruta metodológica inicia con tres elementos interconectados: el planteamiento del problema, los referentes teóricos y el método. A partir de ahí, se deriva el diseño de tareas matemáticas, que conduce a la selección de los informantes y posteriormente a la implementación de las tareas matemáticas. El proceso culmina con la evaluación del pensamiento matemático.

Este diseño se desarrolla a través de las siguientes fases operativas:

La fase de planeación abarca la preparación metodológica inicial, incluyendo la selección de los participantes, la elaboración de las tareas matemáticas y la preparación de los instrumentos de recolección de datos.

Durante la fase de implementación, se ejecutan las actividades diseñadas con los estudiantes, utilizando Python para resolver tareas matemáticas, realizando entrevistas, grupos focales y observaciones de clase.

La fase de análisis comprende el procesamiento de los datos recolectados mediante técnicas cualitativas, identificando patrones y temas emergentes relacionados con el pensamiento matemático.

En la fase de validación se realiza la triangulación de datos obtenidos de diferentes fuentes (entrevistas, observaciones, análisis de trabajos) para asegurar la validez de las interpretaciones.

Finalmente, la fase de socialización implica compartir los resultados con la comunidad educativa, contribuyendo al mejoramiento de las prácticas pedagógicas en matemáticas.

3.3. Escenario de Investigación

La investigación se desarrollará en la Escuela Normal Superior de Quibdó, ubicada en el municipio de Quibdó, capital del departamento del Chocó, Colombia. Esta institución educativa pública, fundada en 1936, atiende a una población estudiantil diversa, predominantemente afrodescendiente e indígena, y ofrece educación desde preescolar hasta formación complementaria. Para este estudio, se enfocará en los niveles de educación básica secundaria y media, los cuales son sexto, octavo y decimo grados

El plantel cuenta con tres laboratorios de computación y conexión a internet, lo que permite la implementación de actividades con Python. Este contexto es ideal para explorar cómo los estudiantes interactúan con la herramienta tecnológica y cómo esta influye en su desarrollo del pensamiento matemático.

3.4. Selección de Informantes Clave

Se seleccionarán como informantes clave a estudiantes de los grados sexto, octavo y décimo de básica secundaria de la Escuela Normal Superior de Quibdó. Estos estudiantes tendrán edades comprendidas entre los 12 y 18 años, residirán cerca de la institución y manifestarán interés por participar voluntariamente en el estudio. La selección de estos grados específicos permitirá obtener una perspectiva amplia del desarrollo del pensamiento matemático a lo largo de diferentes etapas de la educación secundaria.

Se contará con el consentimiento informado de los padres y representantes, y se mantendrá la confidencialidad de los datos personales de todos los informantes para resguardar su integridad. Además, se garantizará que la participación sea voluntaria y que los estudiantes puedan retirarse del estudio en cualquier momento sin consecuencias negativas.

3.5. Consideraciones Éticas

Para la ejecución de la presente investigación, se tendrán en cuenta los siguientes aspectos éticos:

Autorización Institucional: Se solicitará por escrito al rector de la Escuela Normal Superior de Quibdó la autorización para llevar a cabo la implementación del estudio (ver Anexo 5)

Consentimiento Informado: Se enviarán consentimientos informados por escrito a los padres de familia y/o acudientes de los estudiantes seleccionados (ver Anexo 6), especificando que el material recolectado será utilizado exclusivamente con fines académicos y que se respetarán las políticas de seguridad y privacidad de los datos personales.

Anonimato y Confidencialidad: Se garantizará el anonimato de los participantes en todas las etapas de la investigación, utilizando códigos o pseudónimos para referirse a ellos en los informes y publicaciones derivadas del estudio.

Voluntariedad: Se asegurará que la participación de los estudiantes sea voluntaria y que puedan retirarse del estudio en cualquier momento sin que esto implique algún tipo de consecuencia negativa para ellos.

Transparencia: Se brindará información clara y precisa a los estudiantes sobre los objetivos, procedimientos y posibles beneficios de la investigación, respondiendo a cualquier inquietud que puedan tener al respecto.

Manejo de Datos: Los datos recolectados se manejarán de manera confidencial y segura, garantizando que solo los investigadores tendrán acceso a ellos y que serán utilizados únicamente para los fines establecidos en el estudio.

Socialización de Resultados: Se compartirán los resultados de la investigación con la comunidad educativa de la Escuela Normal Superior de Quibdó, respetando siempre la confidencialidad de los participantes y destacando los posibles beneficios que el estudio pueda tener para el mejoramiento de los procesos de enseñanza y aprendizaje de las matemáticas en la institución.

Estas consideraciones éticas se fundamentan en los principios de respeto, beneficencia, no maleficencia y justicia, y buscan garantizar que la investigación se lleve a cabo de manera responsable y transparente, protegiendo en todo momento los derechos y el bienestar de los participantes.

3.6. Técnicas para la Recolección de la Información

Para recolectar la información, se emplearán técnicas cualitativas como entrevistas en profundidad, grupos focales, observaciones y análisis de trabajos realizados por los estudiantes con Python. Estas técnicas permitirán explorar a fondo las vivencias, percepciones y reflexiones de los participantes en torno al fenómeno estudiado.

Entrevistas en Profundidad: Se realizarán entrevistas individuales con los estudiantes para comprender su proceso de pensamiento al resolver tareas matemáticas con Python. Las entrevistas se centrarán en cómo los estudiantes abordan las tareas, las estrategias que utilizan y las dificultades que enfrentan.

Grupos Focales: Se organizarán sesiones grupales donde los estudiantes discutirán sus experiencias y reflexiones sobre el uso de Python en la resolución de tareas matemáticas. Estas discusiones permitirán identificar patrones comunes y divergencias en el pensamiento matemático de los estudiantes.

Observaciones de Clase: Se realizarán observaciones sistemáticas de las clases donde los estudiantes trabajen con Python. Estas observaciones permitirán registrar las interacciones entre los estudiantes, las estrategias de resolución de tareas y las dinámicas de aprendizaje.

Análisis de Trabajos: Se analizarán los trabajos realizados por los estudiantes con Python, evaluando la lógica, eficiencia y creatividad en las soluciones programadas. Este análisis permitirá identificar cómo los estudiantes aplican los conceptos matemáticos en un entorno de programación.

Además, se utilizarán los siguientes instrumentos y herramientas para la recolección y análisis de datos:

Guías de Reconocimiento de Objetos y Significados (GROS) Matemáticos y Computacionales: Estas guías permitirán identificar los objetos y significados matemáticos que los estudiantes manifiestan al resolver tareas con Python (ver Anexos 1 y 2). Además, se utilizarán guías específicas para cada tipo de pensamiento matemático (ver Anexo 4).

Tabla de Características y Objetos Matemáticos: Este instrumento permitirá examinar las soluciones de los estudiantes, identificando cómo utilizan Python para abordar diferentes tipos de pensamiento matemático (numérico, espacial, métrico, aleatorio y variacional).

Google Colaboratory (Colab): Se utilizará como entorno de programación en Python. Esta plataforma permite escribir, ejecutar y compartir código Python en un navegador web, ofreciendo un ambiente interactivo y colaborativo ideal para el trabajo de programación en el contexto educativo.

EduBlocks: Se utilizará como entorno de programación visual en Python. Esta plataforma permite construir programas mediante el ensamblaje de bloques gráficos que se convierten automáticamente en código Python, ofreciendo una interfaz intuitiva que reduce las barreras sintácticas iniciales, ideal para estudiantes que inician su experiencia con la programación en el contexto educativo.

3.7. Uso de Inteligencia Artificial

En el desarrollo de esta investigación, se propone el uso de la inteligencia artificial Gemini como recurso para los estudiantes. Gemini es un asistente virtual que puede responder preguntas, ofrecer explicaciones y brindar orientación a los estudiantes durante el proceso de resolución de las tareas matemáticas con Python. La incorporación de esta herramienta busca enriquecer la experiencia de aprendizaje y promover la autonomía de los estudiantes.

3.8. Evaluación del Pensamiento Matemático

El desarrollo del pensamiento matemático se evaluará a través de las siguientes acciones concretas:

Análisis de los Códigos Python: Se evaluará la lógica, eficiencia y creatividad en las soluciones programadas por los estudiantes.

Entrevistas de Reflexión: Los estudiantes explicarán su proceso de pensamiento y justificarán sus decisiones al resolver las tareas matemáticas.

Observaciones de Clase: Se registrarán las estrategias de resolución de tareas y las discusiones matemáticas que surjan durante las actividades.

Rúbrica de Evaluación: Basada en indicadores específicos para cada tipo de pensamiento matemático (ver Anexo 3), se evaluará el progreso de los estudiantes a lo largo del estudio. Esta tabla de características y objetos matemáticos permitirá examinar sistemáticamente las soluciones desarrolladas por los estudiantes.

Tareas de Transferencia: Se diseñarán tareas matemáticas sin programación para evaluar la transferencia de habilidades de pensamiento matemático adquiridas con Python.

3.9. Análisis de Datos

El análisis de datos en esta investigación se realizará bajo un enfoque cualitativo, utilizando técnicas que permitan comprender en profundidad las experiencias y percepciones de los estudiantes al resolver tareas matemáticas con Python. El proceso de análisis se dividirá en varias etapas, cada una con un propósito específico para garantizar la validez y confiabilidad de los resultados.

3.9.1. Recolección y Organización de Datos

Los datos se recolectarán a través de diversas técnicas, como entrevistas en profundidad, grupos focales, observaciones de clase y análisis de trabajos realizados por los estudiantes. Una vez recolectados, los datos serán organizados y categorizados para facilitar su análisis. El análisis cualitativo se realizó mediante codificación manual y sistemática de los datos, siguiendo los principios del análisis temático propuesto por Braun y Clarke (2006). La codificación y categorización se llevó a cabo utilizando procesadores de texto y hojas de cálculo para organizar y analizar los datos de manera estructurada

3.9.2. Codificación de Datos

La codificación es un proceso en el análisis cualitativo, ya que permite identificar patrones y temas emergentes en los datos. Siguiendo a Salgado (2007), la codificación se realizará en dos etapas:

Codificación Abierta: En esta etapa, se identificarán conceptos y categorías iniciales a partir de los datos recolectados. Por ejemplo, se podrían identificar categorías como "estrategias de resolución de tareas", "dificultades en el uso de Python", "manifestaciones del pensamiento matemático", entre otras.

Codificación Axial: En esta etapa, se establecerán relaciones entre las categorías identificadas en la codificación abierta. Por ejemplo, se podría analizar cómo las estrategias de resolución de tareas se relacionan con las dificultades que los estudiantes enfrentan al usar Python.

La codificación y análisis se realizará utilizando las categorías establecidas en las Guías de Reconocimiento de Objetos y Significados (ver Anexos 1, 2 y 4) y la tabla de características y objetos matemáticos (ver Anexo 3) como marcos de referencia inicial para la identificación de patrones y temas emergentes.

3.9.3. Análisis Temático

El análisis temático es una técnica que permite identificar temas recurrentes en los datos cualitativos. Según Braun y Clarke (2006), el análisis temático implica la identificación, análisis y reporte de patrones (temas) dentro de los datos. En este estudio, se identificarán temas relacionados con el pensamiento matemático manifestado por los estudiantes al resolver tareas con Python. Algunos temas potenciales podrían incluir:

Uso de Python como herramienta de visualización: Cómo los estudiantes utilizan Python para representar gráficamente funciones matemáticas y cómo esto influye en su comprensión de los conceptos.

Transferencia de habilidades matemáticas: Cómo las habilidades desarrolladas con Python se transfieren a la resolución de tareas matemáticas sin el uso de tecnología.

Conflicto entre el pensamiento matemático tradicional y el pensamiento computacional: Cómo los estudiantes integran o diferencian los procesos de pensamiento matemático y computacional al resolver tareas con Python.

3.9.4. Triangulación de Datos

Para asegurar la validez de los resultados, se realizará una triangulación de datos, que consiste en comparar y contrastar los datos obtenidos de diferentes fuentes (entrevistas, observaciones, análisis de trabajos). Según Denzin (1978), la triangulación permite aumentar la credibilidad de los hallazgos al confirmar que los resultados son consistentes a través de múltiples métodos de recolección de datos.

Por ejemplo, si en las entrevistas los estudiantes mencionan que Python les ayuda a visualizar mejor los conceptos matemáticos, esta afirmación se contrastará con las observaciones

de clase y los trabajos realizados, donde se podrá verificar si efectivamente los estudiantes utilizan Python para crear representaciones gráficas y si esto mejora su comprensión.

3.9.5. Interpretación de Resultados

Una vez que se hayan identificado los temas y se haya realizado la triangulación de datos, se procederá a la interpretación de los resultados. Esta interpretación se basará en el marco teórico establecido, en los conceptos de pensamiento matemático, tecnología educativa y el uso de Python en la enseñanza de las matemáticas.

Por ejemplo, si se identifica que los estudiantes tienen dificultades para transferir las habilidades desarrolladas con Python a tareas matemáticas, se podría interpretar que existe una brecha entre el pensamiento computacional y el pensamiento matemático tradicional, lo cual podría tener implicaciones para el diseño de estrategias pedagógicas que integren ambas formas de pensamiento.

3.9.6. Validación de los Hallazgos

Finalmente, los hallazgos serán validados a través de la devolución de resultados a los participantes y a la comunidad educativa. Esto permitirá confirmar que las interpretaciones realizadas son consistentes con las experiencias y percepciones de los estudiantes.

3.10. Caracterización del Pensamiento Matemático

La caracterización del pensamiento matemático es un proceso complejo que requiere un análisis detallado de cómo los estudiantes abordan y resuelven tareas matemáticas, cuando utilizan herramientas tecnológicas como Python. En esta investigación, la caracterización se realizará a través de un enfoque multidimensional, considerando tanto los aspectos cognitivos como los procedimentales del pensamiento matemático.

3.10.1. Procesos de Caracterización

La caracterización del pensamiento matemático se realizará a través de los siguientes procesos, basados en el marco propuesto por Lesh y Doerr (2003), quienes sugieren que el pensamiento matemático puede ser caracterizado a través de la interacción entre representaciones, modelos y contextos:

Diseño de Tareas Específicas: Se diseñarán tareas matemáticas que permitan explorar cada uno de los componentes del pensamiento matemático mencionados anteriormente. Estas tareas serán resueltas por los estudiantes utilizando Python, lo que permitirá observar cómo interactúan con la herramienta tecnológica y cómo aplican los conceptos matemáticos.

Identificación de Representaciones Matemáticas: A través de las Guías de Reconocimiento de Objetos y Significados (GROS), se identificarán las representaciones matemáticas que los estudiantes utilizan al resolver las tareas. Por ejemplo, se analizará cómo los estudiantes representan funciones matemáticas (algebraicamente, gráficamente o numéricamente) y cómo estas representaciones influyen en su comprensión.

Análisis de Modelos y Estrategias: Se analizarán los modelos y estrategias que los estudiantes utilizan para resolver tareas matemáticas con Python. Por ejemplo, se podría identificar que algunos estudiantes prefieren utilizar modelos gráficos para resolver tareas de geometría, mientras que otros prefieren enfoques algebraicos.

Evaluación de la Transferencia de Conocimiento: Se evaluará cómo los estudiantes transfieren las habilidades desarrolladas con Python a tareas matemáticas. Esto permitirá identificar si el uso de Python fomenta un pensamiento matemático más flexible y adaptable.

Descripción de Fortalezas y Debilidades: Se identificarán las fortalezas y debilidades en el pensamiento matemático de los estudiantes, considerando cómo utilizan Python para resolver

tareas matemáticas. Por ejemplo, se podría identificar que algunos estudiantes tienen fortalezas en el pensamiento numérico, pero debilidades en el pensamiento espacial.

El proceso de caracterización se apoyará en los instrumentos desarrollados específicamente para este fin, incluyendo las Guías GROS para diferentes tipos de pensamiento matemático (ver Anexo 4) y la tabla de características y objetos matemáticos (ver Anexo 3).

3.10.2. Implicaciones Pedagógicas

La caracterización del pensamiento matemático tiene un valor descriptivo, que también tiene implicaciones pedagógicas importantes. Los resultados de esta investigación permitirán diseñar estrategias didácticas que integren Python de manera efectiva en la enseñanza de las matemáticas, promoviendo un aprendizaje significativo y contextualizado. Además, se podrán identificar áreas de mejora en el currículo de matemáticas, lo que respecta a la integración de herramientas tecnológicas.

Por ejemplo, si se identifica que los estudiantes tienen dificultades para transferir las habilidades desarrolladas con Python a tareas matemáticas, se podrían diseñar actividades que fomenten la integración del pensamiento computacional y el pensamiento matemático tradicional. Esto podría incluir la creación de tareas que requieran tanto la resolución de problemas con Python como la aplicación de conceptos matemáticos en contextos no tecnológicos.

3.11. Consideraciones sobre la interpretación de procesos cognitivos

El análisis cualitativo de códigos Python para caracterizar el pensamiento matemático implica la interpretación de procesos cognitivos a partir de artefactos producidos por los estudiantes. Si bien este enfoque es metodológicamente apropiado para la investigación fenomenológica y está fundamentado en marcos teóricos establecidos (Schoenfeld, 1992; Mason,

Burton y Stacey, 2010), es importante reconocer las limitaciones inherentes a este tipo de análisis interpretativo.

Las inferencias sobre procesos de pensamiento matemático se basan en la observación de patrones en el código, estrategias de resolución implementadas, y estructuras algorítmicas desarrolladas por los estudiantes. Estas interpretaciones, aunque fundamentadas teóricamente, representan construcciones analíticas del investigador que buscan comprender las manifestaciones observables del pensamiento matemático más que confirmaciones definitivas de procesos cognitivos internos.

La triangulación con entrevistas, observaciones de clase y grupos focales proporciona validación adicional de estas interpretaciones, permitiendo contrastar las inferencias derivadas del análisis de códigos con las perspectivas y experiencias reportadas directamente por los estudiantes.

CAPÍTULO 4

4. ANÁLISIS DE DATOS

4.1. Consideraciones analíticas preliminares

El presente capítulo desarrolla un análisis de los procesos cognitivos y estructuras de pensamiento matemático manifestados por estudiantes de la Escuela Normal Superior de Quibdó al resolver tareas matemáticas mediante el lenguaje de programación Python. Este análisis se fundamenta en una aproximación multidimensional que integra perspectivas cognitivas, didácticas y epistemológicas, y tiene como objetivo caracterizar la naturaleza y evolución del pensamiento matemático en los diferentes niveles educativos estudiados.

4.1.1. Características de los grupos participantes

La investigación se centró en tres grupos específicos: estudiantes de grado sexto (edades entre 11-13 años), estudiantes de grado octavo (edades entre 13-15 años) y estudiantes de grado décimo (edades entre 15-17 años), lo que favorece una visión longitudinal del desarrollo del pensamiento matemático en diferentes grados escolares. Los códigos analizados corresponden a las producciones realizadas durante las sesiones de trabajo con Python, donde se abordaron tareas matemáticas contextualizadas en situaciones cotidianas para los estudiantes.

Tabla 1.

Características de los grupos participantes

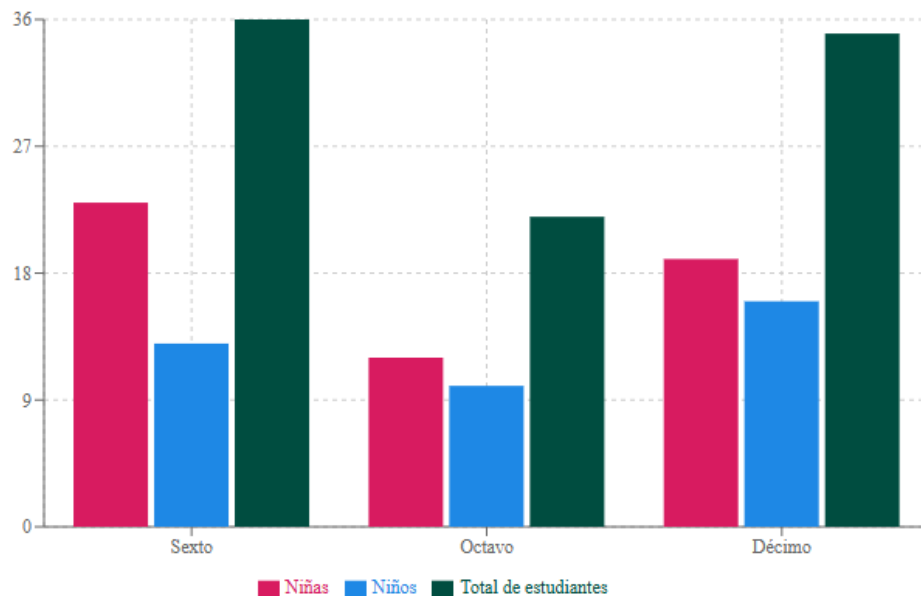
Aspecto	Grado Sexto	Grado Octavo	Grado Décimo
Estudiantes	36 (23 niñas, 13 niños)	22 (12 niñas, 10 niños)	35 (19 niñas, 16 niños)
Infraestructura	Sin computadores disponibles	Computadores para trabajo en pares	Computadores no operativos
Conectividad	Sin acceso a internet	Internet intermitente	Red wifi inutilizable
Adaptación metodológica	Demostración grupal	Trabajo colaborativo limitado	Demostración guiada

Nota: La tabla resume las características demográficas y condiciones técnicas de los grupos que participaron en el estudio, destacando las adaptaciones metodológicas implementadas en cada nivel.

La distribución demográfica de los participantes en el estudio refleja características distintivas en cada nivel educativo. Como se observa en la Figura 5, existe una predominancia femenina en los tres grados analizados, siendo marcada en sexto grado donde las niñas representan casi el doble de la población masculina. Esta distribución por género constituye un elemento contextual relevante para interpretar las manifestaciones del pensamiento matemático identificadas posteriormente.

Figura 5:

Distribución de estudiantes participantes por género y grado académico



Nota: La gráfica muestra la distribución de estudiantes por género en cada uno de los grados participantes, evidenciando una predominancia femenina en los tres niveles educativos analizados.

Las limitaciones técnicas constituyeron un desafío significativo que requirió estrategias diferenciadas. Notable fue el caso del grado décimo, donde la disponibilidad potencial de mejores recursos técnicos se vio frustrada por la falta de conectores para carga de equipos.

4.1.2. Análisis de la conceptualización sobre programación

El primer aspecto analizado fue la comprensión del concepto de programación, revelando una clara progresión evolutiva entre los tres grados:

Tabla 2.

Evolución conceptual sobre programación

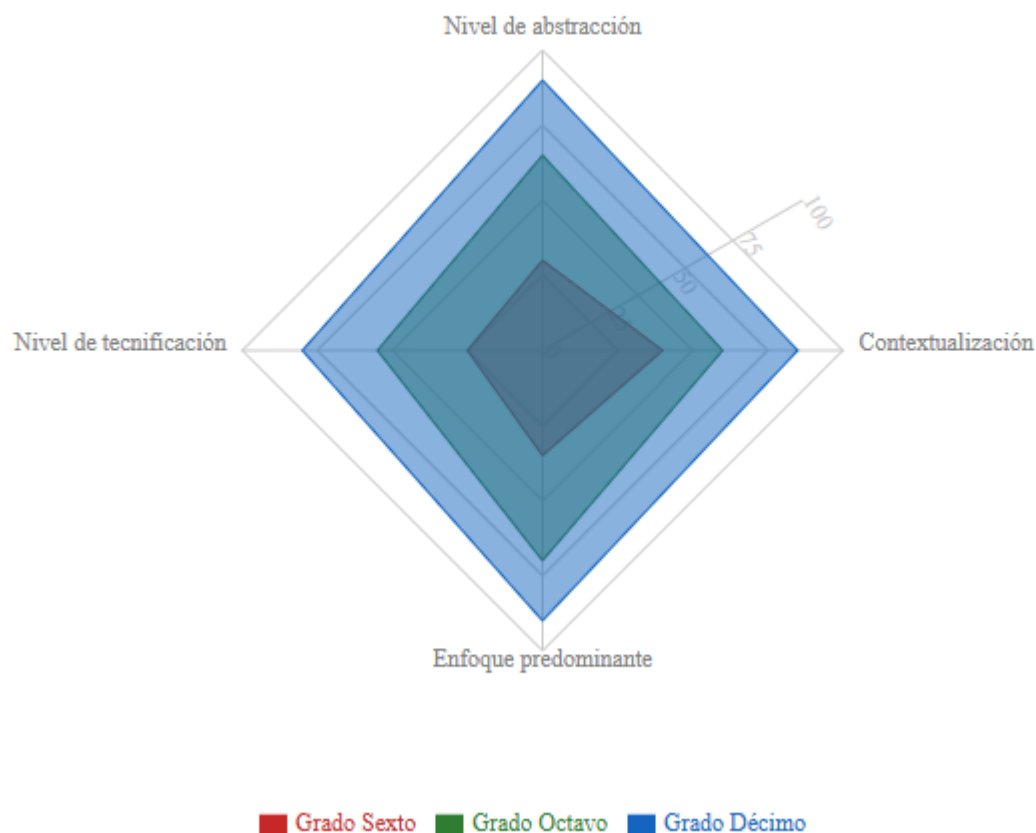
Dimensión	Grado Sexto	Grado Octavo	Grado Décimo
Nivel de abstracción	Concreto: "Programar un juego", "programar datos"	Intermedio: "El proceso de diseñar, escribir y probar"	Avanzado: "El proceso de crear instrucciones para que una computadora realice tareas"
Contextualización	Ejemplos directos: "Como por ejemplo que yo le digo a alguna amiga que si nos vemos a tal hora"	Procedimental: "Sirve para aprender muchas cosas, para realizar problemas"	Sofisticada: "La programación es todo lo que planeamos a futuro", "paso a paso que uno toma para realizar"
Enfoque predominante	Dimensión matemática (35%): vinculación con operaciones	Dimensión técnica (30%): enfoque en diseño y codificación	Dimensión procedimental avanzada (35%): relación con sistemas computacionales
Nivel de tecnificación	Bajo: términos generales sin especificidad técnica	Medio: uso de términos como "variables", "statements"	Alto: referencias a "diseño", "codificación", "instrucciones", "sistema"

Nota: La tabla muestra el desarrollo progresivo de la comprensión del concepto de programación a través de los tres grados analizados, evidenciando transiciones desde definiciones concretas hacia abstracciones más formales.

El análisis de las conceptualizaciones sobre programación revela una clara progresión evolutiva entre los tres grados estudiados. La Figura 6 ilustra esta evolución a través de cuatro dimensiones, donde se evidencia un incremento en los niveles de abstracción, contextualización, enfoque predominante y tecnificación a medida que aumenta el nivel educativo. Resalta el salto cualitativo en el nivel de abstracción, que muestra la transición desde comprensiones concretas en sexto grado hacia conceptualizaciones más formales y abstractas en décimo.

Figura 6:

Evolución del nivel de comprensión conceptual sobre programación en cuatro dimensiones a través de los tres grados analizados



Nota: La gráfica muestra la evolución del nivel de comprensión conceptual sobre programación en las cuatro dimensiones principales analizadas. Los valores representan niveles crecientes de desarrollo en una escala de 0 a 100.

4.1.3. Utilidad percibida y relación con las matemáticas

Las percepciones sobre la utilidad de la programación en el aprendizaje matemático también evidencian un patrón evolutivo claro:

Tabla 3.*Percepción de utilidad para el aprendizaje matemático*

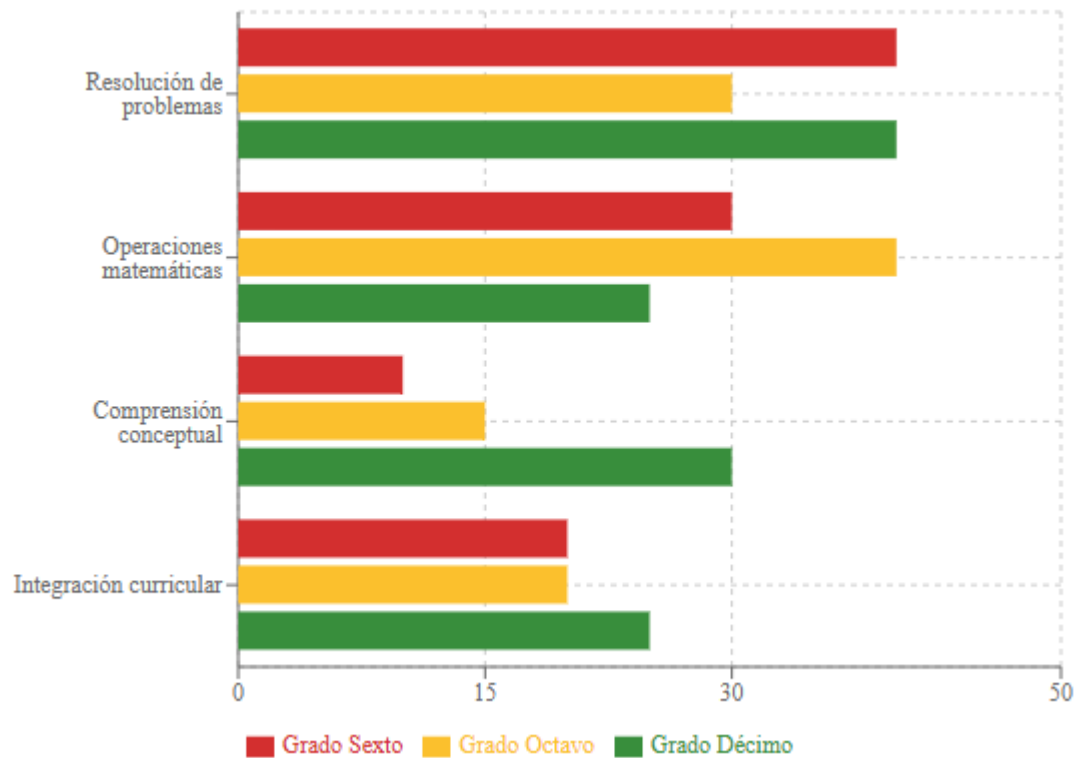
Aspecto	Grado Sexto	Grado Octavo	Grado Décimo
Utilidad principal	Resolución de problemas básicos (40%): "Para resolver problemas matemáticos"	Operaciones matemáticas (40%): "Para realizar operaciones como multiplicaciones, restas, sumas"	Resolución de problemas avanzados (40%): "Para resolver problemas y analizar"
Enfoque secundario	Aprendizaje numérico (30%): "Para aprender con los números enteros"	Resolución problemas (30%): "Herramienta fundamental que permite representar y analizar problemas"	Comprensión conceptual (30%): "Para entender algunos temas, los cuales no los tengo claros"
Operaciones referidas	Principalmente suma y resta	Conjunto completo de operaciones aritméticas	Operaciones avanzadas, representación gráfica, análisis
Integración curricular	General, sin referencia específica	Explícita: con números enteros (20%)	Compleja: sistemas ecuaciones, trigonometría

Nota: La tabla presenta las percepciones predominantes de los estudiantes sobre la utilidad de Python en el contexto educativo matemático, mostrando una evolución desde aplicaciones operativas concretas hacia comprensiones conceptuales más profundas.

Las percepciones de los estudiantes sobre la utilidad de la programación para el aprendizaje matemático también evidencian patrones evolutivos. Como se ilustra en la Figura 7

Figura 7:

Distribución porcentual de percepciones sobre la utilidad de Python para el aprendizaje matemático según categorías y grados



Nota: La gráfica muestra los porcentajes de estudiantes que identificaron cada categoría como utilidad principal de la programación para el aprendizaje matemático, revelando un desplazamiento cualitativo desde operaciones básicas hacia comprensiones conceptuales más profundas.

Esta evolución refleja un desplazamiento desde aplicaciones operativas concretas hacia comprensiones conceptuales más profundas. Aunque la resolución de problemas aparece como utilidad principal en sexto y décimo, la naturaleza de estos problemas muestra una progresión significativa: desde problemas matemáticos elementales en sexto hacia problemas que requieren análisis y modelado en décimo.

4.1.4. Intereses matemáticos específicos

La diversificación de intereses matemáticos manifestados por los estudiantes muestra una clara progresión en complejidad y especificidad:

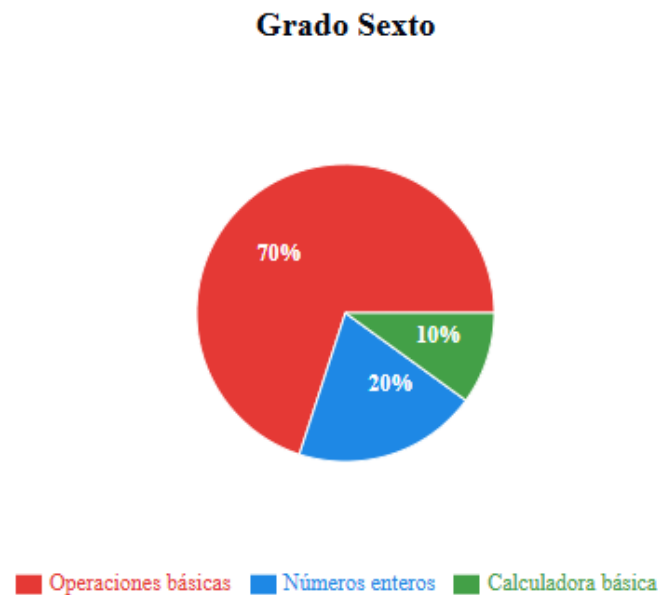
Tabla 4.*Intereses matemáticos por grado*

Grado Sexto	Grado Octavo	Grado Décimo
Operaciones básicas: "La suma y resta"	Operaciones completas: "Suma, resta, multiplicación, división"	Operaciones avanzadas (35%): "Potenciación", "Raíz cuadrada"
Números enteros	Números enteros y decimales	Trigonometría (25%): "Ángulos sexagesimales, coseno, tangente, seno"
Calculadora básica	Aplicación a tareas escolares	Sistemas de ecuaciones (20%): "Ecuaciones polinómicas de grado superior"
Sin especificidad avanzada	Vinculación curricular directa	Geometría (15%): "Problemas de áreas y perímetros" Estadística (5%): "Problemas de estadísticas"

Nota: La tabla evidencia la diversificación progresiva de intereses matemáticos a medida que avanza el nivel educativo, reflejando tanto el desarrollo curricular como la evolución cognitiva de los estudiantes.

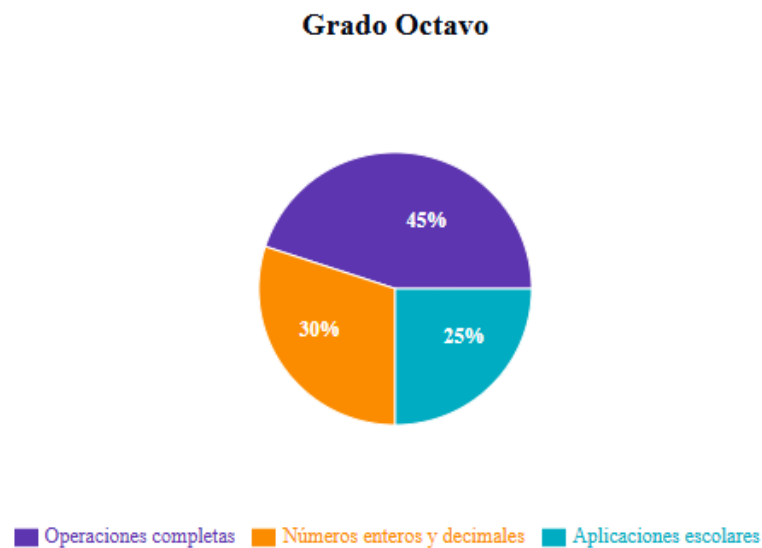
La diversificación de intereses matemáticos manifestados por los estudiantes muestra una progresión notable en complejidad y especificidad. La Figura 8, 9 y 10 visualiza esta evolución a través de los tres niveles educativos.

Figura 8:
Distribución porcentual de intereses matemáticos específicos en estudiantes de sexto grado



Nota: La gráfica muestra la distribución porcentual de intereses matemáticos específicos en sexto grado, evidenciando una marcada concentración en operaciones básicas (70%).

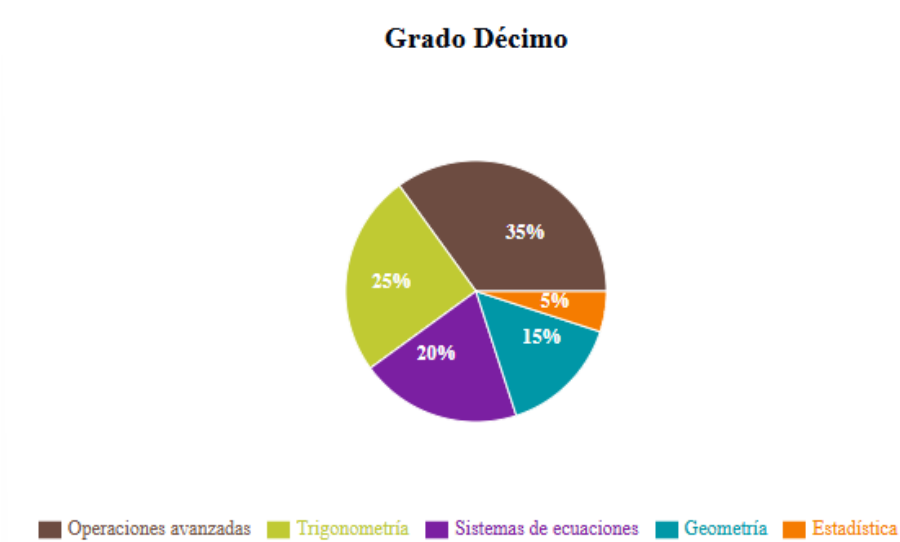
Figura 9:
Distribución porcentual de intereses matemáticos específicos en estudiantes de octavo grado



Nota: La gráfica muestra la distribución porcentual de intereses matemáticos específicos en octavo grado, revelando una mayor diversificación respecto a sexto grado, con emergencia de intereses en aplicaciones curriculares específicas.

Figura 10:

Distribución porcentual de intereses matemáticos específicos en estudiantes de décimo grado



Nota: La gráfica muestra la distribución porcentual de intereses matemáticos específicos en décimo grado, evidenciando una diversificación hacia áreas matemáticas avanzadas como trigonometría, sistemas de ecuaciones y estadística.

Esta diversificación evidencia ampliación significativa del horizonte matemático a medida que avanza el nivel educativo. En sexto grado, los intereses se concentran en operaciones básicas y utilidades elementales como la calculadora. En octavo, aparece una vinculación más directa con el contenido curricular específico (números enteros) y una extensión hacia el conjunto completo de operaciones aritméticas.

En décimo grado, emerge una diversificación hacia áreas matemáticas avanzadas, con intereses específicos en operaciones como potenciación y raíz cuadrada (35%), trigonometría (25%), sistemas de ecuaciones (20%), geometría (15%) y estadística (5%). Esta distribución evidencia un pensamiento matemático más desarrollado, con mayor conciencia de la diversidad de dominios matemáticos y sus aplicaciones potenciales en programación.

4.1.5. Experiencias y dificultades durante la implementación

Las experiencias iniciales reportadas por los estudiantes muestran diferencias cualitativas significativas:

Tabla 5.

Experiencias iniciales con programación

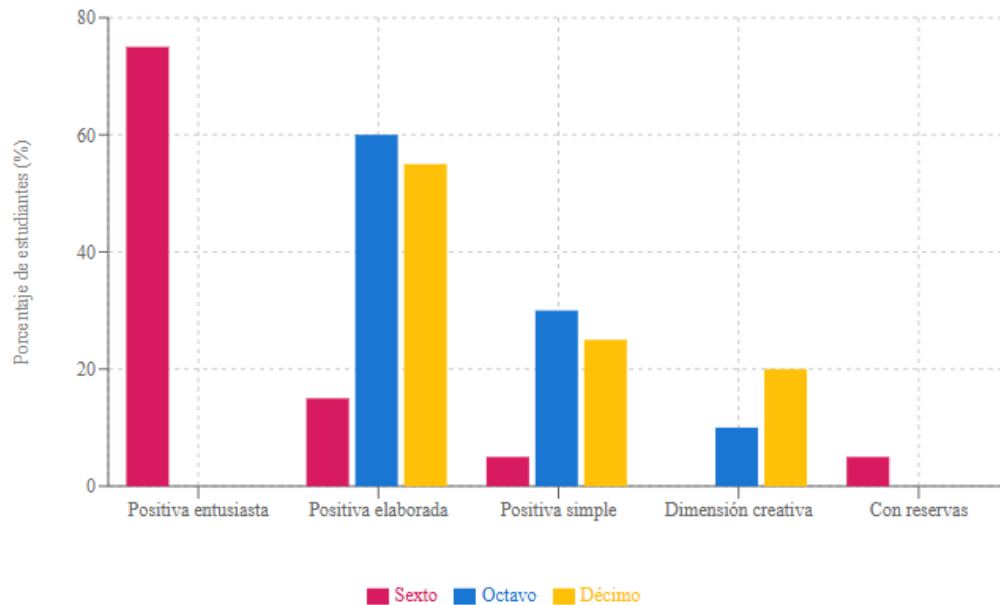
Tipo de Experiencia	Grado Sexto	Grado Octavo	Grado Décimo
Positiva entusiasta	75%: "Feliz y contento", "muy bien, alegre y feliz"		
Positiva elaborada	15%: "Bien porque entendí muchas cosas"	60%: "Me sentí bien porque hicimos nuestro programa de números enteros"	Valoraciones detalladas vinculadas a logros específicos
Positiva simple	5%: "Normal", "bien"	30%: "Bien"	Menor frecuencia de respuestas simples
Dimensión creativa	Ausente	10%: "Me sin creativo ya que puede crear programas"	Referencias a capacidad creativa y generativa
Con reservas	5%: "No tan bien"		

Nota: La tabla categoriza las primeras experiencias de los estudiantes al interactuar con la programación, mostrando un desplazamiento desde expresiones entusiastas simples hacia experiencias vinculadas con logros específicos de aprendizaje.

Las experiencias iniciales reportadas por los estudiantes al interactuar con Python revelan diferencias cualitativas significativas entre niveles educativos. Como se aprecia en la Figura 11.

Figura 11:

Tipos de experiencias iniciales con programación por grado



Nota: La gráfica muestra la distribución porcentual de los tipos de experiencias iniciales con programación reportadas por los estudiantes de cada grado, evidenciando una evolución desde experiencias entusiastas simples hacia valoraciones más elaboradas vinculadas con logros de aprendizaje.

Mientras en sexto grado predominan expresiones entusiastas simples (75%), en octavo y décimo se observa un desplazamiento hacia experiencias positivas elaboradas que vinculan la satisfacción con logros de aprendizaje específicos. Resulta interesante la emergencia de la dimensión creativa en octavo (10%) y su incremento en décimo (20%), evidenciando una progresiva apropiación de la programación como herramienta de expresión creativa.

Las dificultades reportadas por los estudiantes muestran un desplazamiento significativo del foco de atención a medida que avanza el nivel educativo:

Tabla 6.
Evolución de dificultades reportadas

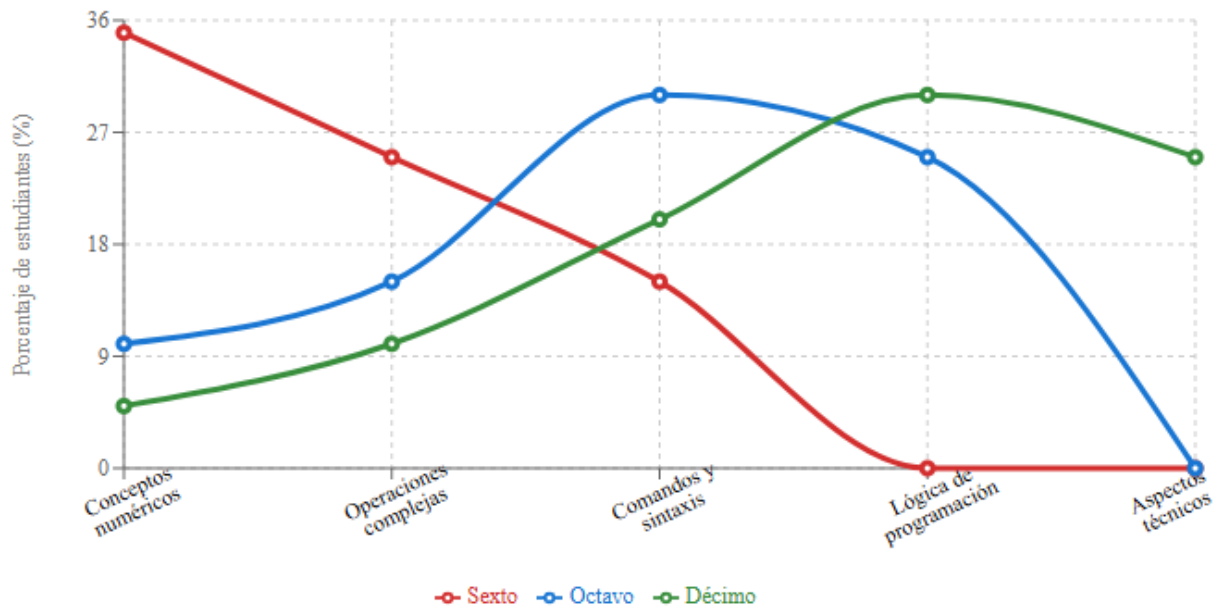
Tipo de Dificultad	Grado Sexto	Grado Octavo	Grado Décimo
Principal	Conceptos numéricos (35%): "Los números más difíciles"	Sintaxis y comandos (30%): "Variables y statements"	Lógica de programación (30%): "Saber cómo entender la lógica, que al mínimo error todo se daña"
Secundaria	Operaciones complejas (25%): "La multiplicación"	Implementación práctica (25%): "Hacer que funcionara la calculadora"	Aspectos técnicos (25%): "Sobre la calculadora y esos números"
Terciaria	Comandos y sintaxis (15%)	Funcionamiento (15%): "Las letras en inglés"	Códigos y comandos (20%): "Memorizarse las mecánicas"
Adicional	Trabajo en equipo (10%)		
Sin dificultad	25%	20%	25%

Nota: La tabla presenta el desplazamiento en el foco de dificultad percibido por los estudiantes, desde obstáculos con conceptos matemáticos hacia desafíos relacionados con aspectos más avanzados de la programación.

Las dificultades reportadas por los estudiantes muestran un desplazamiento del foco de atención a medida que avanza el nivel educativo. La Figura 12 ilustra esta evolución, donde se observa una transición desde dificultades centradas principalmente en conceptos numéricos y operaciones en sexto grado, hacia desafíos relacionados con la sintaxis del lenguaje en octavo grado, y finalmente hacia la comprensión de la lógica de programación en décimo grado. Esta progresión refleja un desplazamiento desde obstáculos en el dominio matemático hacia obstáculos en el dominio computacional, evidenciando la evolución del pensamiento matemático-computacional integrado.

Figura 12:

Evolución de dificultades reportadas por los estudiantes según nivel educativo



Nota: La gráfica muestra la evolución de los tipos de dificultades reportadas por los estudiantes a medida que avanza el nivel educativo, evidenciando un desplazamiento desde dificultades con conceptos matemáticos hacia desafíos relacionados con aspectos más avanzados de la programación.

4.2. Análisis de las manifestaciones del pensamiento matemático en códigos Python

El objetivo central no se limita a identificar patrones de uso de Python, sino a comprender cómo esta herramienta tecnológica interviene en la construcción, representación y transformación del pensamiento matemático, revelando tanto fortalezas emergentes como obstáculos epistemológicos significativos. Esta perspectiva se alinea con lo planteado por Reyes-Santander (2014), quien conceptualiza el pensamiento matemático como un producto complejo derivado de diversos procesos neuropsicológicos que requieren diferentes habilidades tanto internas como externas del individuo.

La aproximación metodológica adoptada se fundamenta en el análisis cualitativo de las producciones de los estudiantes, considerando la corrección de las soluciones y los procesos de

pensamiento reflejados en la estructura, organización y características específicas del código desarrollado.

Consideraciones analíticas para la interpretación de códigos

El análisis detallado de los códigos Python desarrollados por los estudiantes busca identificar manifestaciones del pensamiento matemático a través de las decisiones de programación, estrategias de resolución y estructuras algorítmicas implementadas. Este enfoque analítico se fundamenta en la premisa de que los artefactos producidos por los estudiantes proporcionan ventanas hacia sus procesos de pensamiento matemático subyacentes.

Las interpretaciones presentadas se basan en marcos teóricos establecidos en el campo de la educación matemática y representan análisis fundamentados de patrones observables en el código. Sin embargo, es importante reconocer que estas interpretaciones constituyen construcciones analíticas que, aunque académicamente válidas, requerirían validación adicional mediante protocolos de pensamiento en voz alta o entrevistas cognitivas específicas para confirmación definitiva de los procesos mentales identificados.

El objetivo de este análisis es proporcionar una caracterización sistemática y fundamentada del pensamiento matemático manifestado, reconociendo que las inferencias sobre procesos cognitivos internos representan interpretaciones posibles más que confirmaciones categóricas.

4.2.1. Pensamiento Numérico y Sistemas Numéricos en Sexto Grado

El siguiente código, desarrollado por un estudiante de sexto grado para resolver un problema sobre horarios de lanchas en el puerto de Quibdó, ofrece una ventana hacia las estructuras cognitivas subyacentes al pensamiento numérico en este nivel educativo:

Figura 13:

Código de estudiante de sexto grado sobre horarios de lanchas en Quibdó

Code

```

1 #Start code here
2 tiempo_laancha1 = 4
3 tiempo_laancha2 = 6
4 hora_inicial = 6
5 multiplo_laancha1 = [4, 8, 12, 16, 20, 24]
6 multiplo_laancha2 = [6, 12, 18, 24]
7 mcm = 12
8 proxima_concidencia = hora_inicial + mcm
9 print("6:00pm")
10

```

Análisis detallado de estructuras cognitivas y representaciones:

- **Representación del MCM mediante listas explícitas:** El estudiante elige representar los múltiplos de cada número (4 y 6) mediante listas explícitas (`multiplo_laancha1` y `multiplo_laancha2`). Esta decisión revela un enfoque concreto y visual para la comprensión del MCM, donde el concepto se materializa como la intersección observable entre dos secuencias numéricas. En términos del pensamiento numérico descrito por el MEN (1998), esto demuestra una comprensión basada en la manipulación directa de números y en la identificación de patrones de repetición, característica del pensamiento numérico en desarrollo.

- **Ausencia de algoritmo para generar múltiplos:** A diferencia de un enfoque algorítmico que generaría los múltiplos mediante un proceso iterativo, el estudiante construye las listas manualmente. Esta decisión refleja que, si bien existe comprensión del concepto de múltiplo, aún no se ha desarrollado completamente la capacidad de abstraer este concepto en un procedimiento generalizable. Según Mason, Burton y Stacey (2010), este comportamiento es la fase exploratoria del pensamiento matemático, donde los ejemplos específicos preceden a las generalizaciones.
- **Identificación visual del MCM sin proceso algorítmico:** El estudiante determina que 12 es el MCM al observar que es el primer valor común en ambas listas, evidenciando una comprensión conceptual correcta del MCM como el menor múltiplo compartido. Sin embargo, no implementa un proceso algorítmico para encontrarlo (como podría ser comparar elementos o utilizar el algoritmo de Euclides). Esto evidencia lo que Schoenfeld (1992) describe como comprensión cualitativa de un concepto sin la correspondiente estructura procedimental formalizada.
- **Traducción directa del contexto al modelo matemático:** La asignación de variables como `tiempo_lancha1` y `hora_inicial` muestra capacidad para traducir elementos de la tarea contextualizada a variables matemáticas, estableciendo un primer nivel de abstracción. Esta traducción, según Flores Samaniego (2017), constituye un paso en el desarrollo del pensamiento matemático, pues implica la capacidad de identificar y representar simbólicamente cantidades relevantes dentro de un contexto específico.

- **Aplicación contextual del MCM:** La línea `proxima_concidencia = hora_inicial + mcm` demuestra comprensión de la aplicación práctica del MCM en el contexto del problema. El estudiante reconoce que, partiendo de un tiempo inicial, el MCM representa el intervalo de tiempo necesario para que ambas lanchas vuelvan a coincidir. Esta aplicación refleja lo que el MEN (1998) describe como capacidad para identificar y utilizar relaciones numéricas en situaciones concretas.
- **Transformación implícita del formato de hora:** Aunque el cálculo numérico da como resultado 18 (6 + 12), el estudiante imprime "6:00pm", que evidencia que realizó mentalmente la conversión de 18 horas al formato de 12 horas. Esta operación, no explícita en el código, revela un proceso cognitivo complementario que refleja familiaridad con diferentes sistemas de representación del tiempo, aspecto relevante del pensamiento métrico según el MEN (1998).
- **Limitaciones y obstáculos identificados:** Restricción a valores específicos: La solución está completamente ligada a los valores específicos del problema (4 y 6) y no constituye un algoritmo general para calcular el MCM de cualesquiera dos números. Esta limitación evidencia lo que Kilpatrick et al. (2001) describen como una etapa inicial en el desarrollo de la fluidez procedimental, donde las soluciones son específicas y no generalizables.
- **Ausencia de procesos de verificación:** El código no incluye ningún mecanismo para verificar que 12 sea efectivamente el MCM, ni comprueba que los valores en las listas sean realmente múltiplos de 4 y 6 respectivamente. Esta ausencia de

verificación representa una limitación en lo que Schoenfeld (1992) identifica como procesos metacognitivos esenciales para el pensamiento matemático avanzado.

- **Desarrollo manual de listas de múltiplos:** La construcción manual de las listas de múltiplos, en lugar de generarlas mediante un bucle o fórmula, sugiere que el estudiante aún no ha desarrollado completamente la capacidad de generalizar procesos repetitivos mediante estructuras algorítmicas. Esto refleja una etapa transitoria en el desarrollo del pensamiento matemático según lo descrito por Mason, Burton y Stacey (2010).
- **Comprensión intuitiva sin formalización matemática:** El estudiante exhibe una comprensión intuitiva correcta del MCM, pero sin expresarla a través de definiciones o propiedades formales. Esta característica se alinea con lo que Sánchez (1995) identifica como una fase intermedia en el desarrollo del pensamiento, donde los conceptos se comprenden operativamente pero no se han formalizado completamente.

Esta aproximación al problema del MCM revela un pensamiento matemático en desarrollo que, si bien muestra comprensión conceptual, aún se encuentra en proceso de consolidar aspectos procedimentales y formales. Las características identificadas proporcionan información valiosa sobre cómo los estudiantes de sexto grado conceptualizan y operacionalizan nociones del pensamiento numérico a través de Python.

4.2.2. Pensamiento Espacial y Sistemas Geométricos en Sexto Grado

El código siguiente, orientado a resolver un problema sobre áreas y perímetros en el mercado de Quibdó, revela estructuras relacionadas con el pensamiento espacial:

Figura 14:

Código sobre área y perímetro utilizado por un estudiante de sexto grado.

Code

```

1  #Start code here
2  equipo_roblox = 3 * 2
3  print(equipo_roblox)
4  print("se necesita 6 metros cuadrados")
5  perimetro = 3 + 2 + 3 + 2
6  print(perimetro)
7  print(" se necesita 10 metros cuadrados")
8  equipo_roblox = 8000 * 6
9  print(equipo_roblox)
10 equipo_frefire = 2000 * 10
11 print(equipo_frefire)
12 total = equipo_roblox + equipo_frefire
13 print(total)
14 print("gasto en total 68000 pesos")
15

```

Análisis detallado de procesos cognitivos y manifestaciones del pensamiento espacial:

- Denominación idiosincrática pero significativa de variables:** El uso de términos como "equipo_roblox" para representar el área del puesto revela un proceso cognitivo: la apropiación personal de conceptos abstractos mediante su asociación con elementos culturalmente significativos para el estudiante. Esta estrategia, lejos de ser anecdótica, constituye lo que Sánchez (1995) describe como un mecanismo de "anclaje cognitivo", donde conocimientos nuevos se vinculan a estructuras cognitivas preexistentes significativas para el aprendiz. Roblox, como plataforma de juegos populares entre niños y adolescentes, proporciona un referente familiar que facilita la manipulación cognitiva del concepto más abstracto de área.

Cálculo directo del área rectangular: La línea `equipo_roblox = 3 * 2` muestra comprensión operativa de la fórmula del área del rectángulo ($\text{base} \times \text{altura}$). Esta implementación

directa revela lo que el MEN (1998) identifica como comprensión del significado de las operaciones básicas aplicadas a magnitudes espaciales, elemento del pensamiento espacial en este nivel educativo. El estudiante no necesita consultar o derivar la fórmula, lo que indica que este conocimiento ya está interiorizado.

Cálculo explícito del perímetro mediante suma de lados: En lugar de aplicar la fórmula general del perímetro de un rectángulo ($2 \times (\text{base} + \text{altura})$), el estudiante suma individualmente cada uno de los cuatro lados: $\text{perímetro} = 3 + 2 + 3 + 2$. Esta aproximación revela una comprensión concreta y directa del perímetro como "suma de todas las longitudes de los lados", sin abstracción de la propiedad de igualdad de lados opuestos en un rectángulo. Según el modelo de Van Hiele (1986) para el desarrollo del pensamiento geométrico, esto sitúa al estudiante en el Nivel 1 (visual) transitando hacia el Nivel 2 (descriptivo/analítico), donde reconoce componentes, pero aún no ha formalizado completamente sus propiedades.

Error conceptual en unidades de medida: La línea que imprime "se necesita 10 metros cuadrados" después de calcular el perímetro revela un obstáculo epistemológico significativo: la confusión entre unidades lineales (metros para perímetro) y unidades de área (metros cuadrados). Este error conceptual, común en el aprendizaje de la geometría según González (2001), indica que el estudiante aún no ha logrado la comprensión dimensional de las magnitudes geométricas, aspecto crítico del Pensamiento Espacial y Métrico.

Traducción correcta entre magnitudes geométricas y costos: Las líneas $\text{equipo_roblox} = 8000 \times 6$ y $\text{equipo_frefire} = 2000 \times 10$ muestra comprensión de la proporcionalidad directa entre magnitudes geométricas (área, perímetro) y sus costos asociados (precio por metro cuadrado, precio por metro lineal). Esta traducción entre dominios conceptuales diferentes (geométrico y

económico) evidencia lo que Kilpatrick et al. (2001) identifican como “competencia estratégica”, la habilidad para formular, representar y resolver problemas matemáticos en diferentes contextos.

Reasignación de variables sin conservación semántica: La variable `equipo_roblox` se utiliza primero para representar el área (6 metros cuadrados) y posteriormente para representar el costo de la lona ($8000 * 6 = 48000$ pesos). Esta reasignación, sin cambio de nombre, revela una comprensión aún incipiente del concepto de variable como entidad con significado semántico consistente, aspecto del pensamiento variacional según el MEN (1998).

Integración de cálculos parciales para obtener el total: La línea `total = equipo_roblox + equipo_frefire` muestra comprensión de la aditividad de costos y la capacidad de integrar resultados parciales para obtener un resultado final, evidenciando lo que Mason, Burton y Stacey (2010) describen como capacidad de síntesis en el proceso de resolución de problemas.

Análisis de patrones cognitivos subyacentes:

Pensamiento concreto dominante: El código revela un pensamiento predominantemente concreto, centrado en operaciones directas sobre valores específicos más que en generalizaciones o abstracciones. Esta característica, según Piaget e Inhelder (1969), es congruente con la etapa de operaciones concretas típica de este grupo etario, donde los estudiantes pueden realizar operaciones lógicas sobre objetos concretos, pero manifiestan dificultades con conceptos abstractos.

Fragmentación procesal con integración emergente: El problema se aborda mediante pasos discretos (calcular área, calcular perímetro, calcular costos individuales, calcular costo total), con una estructura secuencial que muestra capacidad emergente para integrar procesos en un flujo coherente. Esta organización refleja lo que Schoenfeld (1992) describe como desarrollo progresivo de estrategias de resolución de problemas.

Apropiación contextual de conceptos matemáticos: La denominación de variables con términos culturalmente significativos (equipo_roblox, equipo_frefire) evidencia un mecanismo cognitivo importante: la tendencia a contextualizar conceptos abstractos en marcos referenciales familiares facilita su manipulación cognitiva. Este proceso de apropiación personal constituye, según Freudenthal (1991), un aspecto de la “matematización horizontal”, donde la realidad se transforma gradualmente en conceptos matemáticos.

Distinción emergente entre magnitudes y sus medidas: Aunque con errores conceptuales (confusión entre metros y metros cuadrados), el código revela un esfuerzo por distinguir entre diferentes tipos de magnitudes geométricas (unidimensionales y bidimensionales) y sus correspondientes unidades de medida. Esta distinción, según González (2001), es esencial para el desarrollo del pensamiento espacial y métrico.

Transferencia interdominios: El estudiante muestra capacidad para transferir conceptos entre diferentes dominios (geométrico $\rightarrow$ económico), evidenciando desarrollo de lo que Gardner (1983) denomina “pensamiento flexible”, la capacidad de aplicar conocimientos de un área a otra.

El análisis de este código revela un Pensamiento Espacial y Geométrico en desarrollo, con comprensión operativa de conceptos como área y perímetro, pero con limitaciones en la formalización conceptual y en la comprensión dimensional de magnitudes. La estrategia de resolución evidencia procesos cognitivos propios de la transición entre el pensamiento concreto y el pensamiento formal, con mecanismos idiosincráticos de apropiación conceptual que merecen ser reconocidos como parte legítima del proceso de construcción del conocimiento matemático.

4.2.3. Pensamiento Métrico y Sistemas de Medidas en Sexto Grado

El siguiente código, desarrollado para resolver un problema sobre conversión de unidades en el contexto del mercado de pescado, ilustra aspectos del pensamiento métrico:

Figura 15:

Código de conversión de unidades (kilogramos y gramos) usado por un estudiante de sexto grado.

Code

```

1 #Start code here
2 peso_kg = 2.5
3 tama_C3_B1o_porcion = 250
4 precio_porcion_g = 5000
5 num_pescado = 15
6 peso_g = peso_kg * 1000
7 print(peso_g)
8 NUM_PORCIONES = peso_g / tama_C3_B1o_porcion
9 print(NUM_PORCIONES)
10 DINERO_POR_PESCADO = NUM_PORCIONES * 2
11 PRECIO_PORCION = 1 * 2
12

```

Análisis exhaustivo de estructuras representacionales y operaciones cognitivas:

Representación coherente de magnitudes con unidades implícitas: El código revela una estructura cognitiva: la capacidad para representar magnitudes con sus unidades implícitas en el nombre de las variables (`peso_kg`, `tama_C3_B1o_porcion`). Esta práctica muestra conciencia sobre la importancia de las unidades en el manejo de magnitudes físicas, aspecto del Pensamiento Métrico según el MEN (1998). La distinción entre kilogramos y gramos evidencia comprensión de la existencia de diferentes escalas para una misma magnitud física.

Conversión explícita entre unidades de masa: La línea `peso_g = peso_kg * 1000` implementa correctamente la conversión de kilogramos a gramos, mostrando comprensión operativa de la relación entre estas unidades de medida. Esta operación evidencia lo que González (2001) identifica como comprensión de las relaciones multiplicativas entre unidades de un mismo

sistema de medida, componente del Pensamiento Métrico. El factor de conversión (1000) se aplica correctamente, lo que evidencia conocimiento específico sobre la equivalencia entre estas unidades.

Manipulación de unidades heterogéneas en cálculos compuestos: El cálculo de NUM__PORCIONES mediante la división $\text{peso_g} / \text{tama_C3_B1o_porcion}$ muestra capacidad para manipular unidades heterogéneas (gramos totales divididos por gramos por porción) para obtener una magnitud adimensional (número de porciones). Esta operación, según Lehrer (2003), evidencia un nivel avanzado en el desarrollo del pensamiento sobre unidades de medida, donde se comprende que la división entre magnitudes del mismo tipo produce resultados adimensionales.

Distinción tipográfica entre datos y resultados calculados: El uso sistemático de mayúsculas para variables que representan resultados calculados (NUM__PORCIONES, DINERO_POR_PESCADO) revela un mecanismo metacognitivo: la necesidad de distinguir visualmente entre datos iniciales y resultados derivados. Esta práctica evidencia desarrollo de lo que Schoenfeld (1992) denomina como “control ejecutivo”, la capacidad para monitorear y organizar el propio proceso de pensamiento durante la resolución de problemas.

Estructura de cálculo incompleta pero significativa: Aunque las últimas líneas del código ($\text{DINERO_POR_PESCADO} = \text{NUM_PORCIONES} * 2$ y $\text{PRECIO_PORCION} = 1 * 2$) contienen operaciones aparentemente incorrectas o incompletas para el contexto del problema, revelan un intento de continuar la cadena de cálculos desde las porciones hacia valores monetarios. Esta estructura, aunque imperfecta, muestra comprensión de la necesidad de transformaciones secuenciales para resolver el problema completo.

Problema de codificación como obstáculo técnico: La variable `tama_C3_B1o_porcion` muestra un problema típico de codificación de caracteres especiales (la “ñ” se ha convertido en

“_C3_B1”), lo que podría constituir un obstáculo técnico que interfiere con la implementación correcta del algoritmo. Este fenómeno ilustra cómo aspectos técnicos de la programación pueden convertirse en obstáculos para la expresión del pensamiento matemático, especialmente para estudiantes en etapas iniciales de familiarización con entornos de programación.

Análisis de estructuras cognitivas subyacentes:

Construcción de cadena de transformaciones entre dominios: El código revela un intento de establecer una cadena de transformaciones que conecta diferentes dominios conceptuales: físico (peso) → cuantitativo (número de porciones) → económico (valor monetario). Esta estructura cognitiva, según González (2001), es característica del pensamiento matemático avanzado, donde se establecen conexiones entre diferentes sistemas de representación.

Comprensión operativa de proporcionalidad directa: La determinación del número de porciones mediante división evidencia comprensión de la proporcionalidad directa como relación matemática. Esta comprensión, según el MEN (1998), constituye un componente tanto del pensamiento numérico como del pensamiento variacional.

Conservación numérica en transformaciones de unidades: La conversión de kilogramos a gramos muestra comprensión de la “conservación numérica”, principio cognitivo identificado por Piaget e Inhelder (1969), donde se reconoce que la cantidad se mantiene invariante, aunque cambie la forma de representarla o medirla.

Aproximación procedimental secuencial: El código muestra una aproximación eminentemente procedimental y secuencial a la resolución del problema, donde cada línea representa un paso discreto en la cadena de razonamiento. Esta estructura, según Mason, Burton y

Stacey (2010), es característica de las etapas iniciales del desarrollo del pensamiento matemático, donde los problemas se abordan mediante secuencias de operaciones discretas.

Distinción incipiente entre datos y metadatos: El uso diferenciado de tipografía (mayúsculas vs. Minúsculas) indica comprensión emergente de la distinción entre datos (información dada) y metadatos (información sobre la información), aspecto que Schoenfeld (1992) identifica como componente esencial del desarrollo metacognitivo en matemáticas.

En síntesis, este código revela Pensamiento Métrico en desarrollo que muestra comprensión operativa de conceptos como unidades de medida, conversiones y relaciones proporcionales. La estructura cognitiva evidenciada se caracteriza por aproximaciones secuenciales y procedimentales, con desarrollo progresivo de capacidades metacognitivas para organizar y monitorear el proceso de resolución. Las dificultades identificadas se relacionan principalmente con la implementación completa de la cadena de transformaciones necesarias para resolver el problema íntegramente, lo que revela áreas específicas para intervención didáctica.

4.2.4. Pensamiento Espacial y Teorema de Pitágoras en Octavo Grado

El código desarrollado por un estudiante de octavo grado para aplicar el Teorema de Pitágoras en construcciones de palafitos revela estructuras cognitivas más avanzadas, pero con inconsistencias:

Figura 16:

Código de estudiante de octavo grado aplicando el Teorema de Pitágoras en el contexto de palafitos.

Code

```

1 #Start code here
2 BASE_POSTE = 4
3 ALTURA_POSTE = 3
4 SUMA__CUADRADOS = BASE_CUADRADO + ALTURA_CUADRADO
5 LONGITUD__POSTE = RAIZ_CUADRADA_DE_SUMA__CUADRADOS
6 LADO_1 = 3
7 LADO_2 = 4
8 LADO_3 = 5
9 LADO_1__CUADRADO = BASE_POSTE
10 ALTURA_CUADRADO = ALTURA_POSTE * ALTURA_POSTE
11

```

Análisis exhaustivo de estructuras conceptuales y manifestaciones del pensamiento geométrico:

Formalización tipográfica de variables constantes: El uso sistemático de mayúsculas y el patrón de nombrado estructurado (BASE_POSTE, ALTURA_POSTE) ponen de manifiesto un intento deliberado de adoptar convenciones formales de programación. Esta práctica, según Flores Samaniego (2017), constituye un indicio del desarrollo de lo que denomina “pensamiento matemático formal”, donde importa tanto el contenido conceptual como la forma de representarlo según convenciones establecidas. Esta formalización representa un avance significativo respecto al nombrado más idiosincrático observado en los estudiantes de sexto grado.

Conceptualización estructurada del Teorema de Pitágoras: Las líneas $SUMA_CUADRADOS = BASE_CUADRADO + ALTURA_CUADRADO$ y $LONGITUD_POSTE = RAIZ_CUADRADA_DE_SUMA_CUADRADOS$ revelan una comprensión conceptual clara y estructurada del Teorema de Pitágoras, reconociendo explícitamente sus componentes: (1) elevación al cuadrado de los catetos, (2) suma de estos

cuadrados, y (3) extracción de la raíz cuadrada para obtener la hipotenusa. Esta estructuración, según el MEN (1998), evidencia un nivel avanzado de pensamiento espacial donde se comprenden relaciones geométricas.

Reconocimiento explícito de la terna pitagórica (3,4,5): Las líneas que definen LADO_1 = 3, LADO_2 = 4 y LADO_3 = 5 demuestran conocimiento específico de esta terna pitagórica clásica. Este conocimiento, según Mason, Burton y Stacey (2010), representa un caso de “generalización a través de ejemplos específicos”, donde casos particulares conocidos proporcionan anclaje cognitivo para la comprensión de relaciones matemáticas generales. La identificación de esta terna particular revela familiaridad con patrones numéricos importantes en geometría.

Cálculo explícito del cuadrado mediante multiplicación directa: La línea $ALTURA_CUADRADO = ALTURA_POSTE * ALTURA_POSTE$ muestra comprensión operativa del significado de “elevar al cuadrado” como multiplicación de un número por sí mismo. Esta implementación, según Schoenfeld (1992), evidencia conexión entre la representación simbólica de una operación (x^2) y su significado operativo ($x \times x$), aspecto del pensamiento matemático.

Nomenclatura descriptiva para operaciones matemáticas: El uso de variables como RAIZ_CUADRADA_DE_SUMA__CUADRADOS revela un fenómeno cognitivo interesante: la tendencia a utilizar el código como instrumento de cálculo y como medio de expresión conceptual. Este fenómeno se manifiesta, igualmente, en el trabajo de los estudiantes (Kucheman, 1978; Castro, Godino, y Rivas, 2011) en el cual las tienen un doble papel: recordar el nombre de la variable y designar la variable. El estudiante utiliza el nombre de la variable para describir la operación matemática que representa, creando lo que podríamos denominar “código

autoexplicativo”. Esta práctica, aunque ineficaz desde el punto de vista computacional, tiene valor metacognitivo al hacer explícito el razonamiento subyacente.

Inconsistencias y obstáculos epistemológicos identificados:

Variables no definidas previamente: El código intenta utilizar variables como `BASE_CUADRADO` y `ALTURA_CUADRADO` sin haberlas definido previamente. Este error revela un obstáculo epistemológico importante: la dificultad para distinguir entre el concepto matemático abstracto (el cuadrado de la base) y su implementación procedimental en un lenguaje de programación (que requiere cálculo y asignación explícitos). Este fenómeno ilustra lo que Schoenfeld (1992) identifica como brecha entre el conocimiento conceptual y el conocimiento procedimental.

Discontinuidad en la cadena lógica de operaciones: El código muestra una secuencia incompleta y no siempre coherente de operaciones, donde algunas variables se definen, pero no se utilizan (`LADO_1`, `LADO_2`, `LADO_3`), mientras que otras se utilizan sin definir previamente (`BASE_CUADRADO`, `ALTURA_CUADRADO`). Esta discontinuidad, según Mason, Burton y Stacey (2010), refleja dificultades en el desarrollo de la coherencia lógica del pensamiento matemático.

Inconsistencia en la implementación del mismo tipo de operación: El estudiante calcula explícitamente el cuadrado de la altura mediante multiplicación (`ALTURA_CUADRADO = ALTURA_POSTE * ALTURA_POSTE`), pero intenta utilizar directamente `BASE_CUADRADO` sin cálculo explícito similar. Esta inconsistencia revela lo que Kilpatrick et al. (2001) describen como desarrollo parcial de la “fluidez procedimental”, donde los procedimientos se aplican de manera inconsistente o incompleta.

Ausencia de implementación funcional completa: El código define una estructura conceptual pero no llega a implementar un algoritmo funcional completo que calcule realmente la longitud del poste. Este fenómeno ilustra lo que Zepa (2003) identifica como brecha entre la comprensión conceptual y la capacidad de implementación práctica, aspecto crítico en el desarrollo del pensamiento matemático.

Análisis de procesos cognitivos subyacentes:

Abstracción matemática en desarrollo: El código evidencia un nivel de abstracción matemática significativamente mayor que en los ejemplos de sexto grado. El estudiante manipula valores concretos, e intenta representar estructuras conceptuales abstractas (como el Teorema de Pitágoras), aunque con implementación incompleta. Este desarrollo, según Piaget e Inhelder (1969), es característico de la transición hacia el pensamiento formal.

Metarrepresentación emergente: El uso de variables con nombres que describen operaciones (`RAIZ_CUADRADA_DE_SUMA__CUADRADOS`) revela el desarrollo de capacidades meta-representacionales, donde el código representa valores, operaciones y conceptos. Esta capacidad, según Schoenfeld (1992), es fundamental para el desarrollo del pensamiento matemático avanzado.

Formalización conceptual adelantada a la implementación procedural: El código muestra un fenómeno cognitivo interesante: el estudiante ha formalizado conceptualmente el Teorema de Pitágoras (comprende sus componentes y estructura) pero encuentra dificultades para implementarlo procedimentalmente en Python. Esta brecha, según Mason, Burton y Stacey (2010), es común en el desarrollo del pensamiento matemático, donde la comprensión conceptual y la implementación procedural pueden evolucionar a ritmos diferentes.

Estructuración jerárquica del conocimiento: La descomposición del Teorema de Pitágoras en sus componentes (cuadrados de catetos, suma, raíz cuadrada) evidencia capacidad para estructurar jerárquicamente el conocimiento matemático. Esta estructuración, según Kilpatrick et al. (2001), es característica del desarrollo de la “comprensión conceptual” en matemáticas.

En términos del modelo de Van Hiele (1986) para el desarrollo del pensamiento geométrico, este código sitúa al estudiante en transición entre el Nivel 2 (descriptivo/analítico) y el Nivel 3 (abstracto/relacional): comprende propiedades formales del triángulo rectángulo y las relaciones pitagóricas, pero muestra dificultades en la organización lógica formal de estas relaciones en un sistema operativo coherente.

4.2.5. Pensamiento Variacional y Sistemas de Datos en Décimo Grado

El siguiente código, desarrollado por un estudiante de décimo grado para analizar datos turísticos en Quibdó, ilustra estructuras más avanzadas de pensamiento variacional y aleatorio:

Figura 17:

Código utilizado por un estudiante de décimo grado para convertir series temporales (temperatura y precipitaciones).

Code

```

1 #Start code here
2 meses = ["enero", "febrero", "marzo", "abril", "mayo", "junio"]
3 temperatura_C_22 = [28, 29, 27, 30, 29, 28]
4 precipitaciones = [650, 700, 800, 750, 600, 550]
5 temperatura_F_22 = []
6 for temp in temperatura_C_22:
7     temp_F_22 = temp * 9/5 + 32
8     temperatura_F_22.append(temp_F_22)
9 variaciones = []
10 for i in :
11     pass
12

```

Análisis detallado de estructuras cognitivas y representaciones avanzadas:

Estructuración multidimensional de series temporales: El código comienza con la creación de tres listas paralelas y coordinadas: meses, temperatura_C_22 y precipitaciones. Esta estructura refleja una conceptualización sofisticada de series temporales multivariadas, donde diferentes magnitudes (temperatura, precipitación) se organizan en secuencias paralelas asociadas con una misma escala temporal (meses). Según el MEN (1998), esta organización evidencia un nivel avanzado del Pensamiento Variacional, donde se comprenden y manipulan relaciones entre variables que cambian conjuntamente.

Nomenclatura estructurada y semánticamente significativa: El patrón de nombrar variables (temperatura_C_22, temperatura_F_22) revela una estructuración cognitiva avanzada, donde los nombres incorporan el concepto (temperatura), metadatos como la unidad de medida (C, F) e información contextual (22, posiblemente indicando el año o versión). Esta práctica, según Schoenfeld (1992), evidencia desarrollo de lo que denomina “conocimiento condicional”, la comprensión de ‘cuándo’ y ‘por qué’ utilizar determinados conceptos o procedimientos.

Implementación funcional de transformación sistemática: El bucle ‘for’ implementado para convertir todas las temperaturas de Celsius a Fahrenheit representa una estructura algorítmica significativamente más avanzada que en los niveles anteriores. Esta estructura, según Wing (2006), evidencia desarrollo del “pensamiento computacional” integrado con el pensamiento matemático, donde se comprende que la misma operación puede aplicarse sistemáticamente a cada elemento de una colección.

Comprensión operativa de transformación de escalas: La implementación correcta de la fórmula para convertir temperaturas ($\text{temp} * 9/5 + 32$) demuestra comprensión conceptual y operativa de las transformaciones lineales entre escalas de medida. Esta comprensión, según

González (2001), representa un aspecto avanzado del Pensamiento Métrico, donde se entienden y manipulan relaciones funcionales entre diferentes sistemas de medición.

Construcción dinámica de colecciones con ‘append()’: El uso del método append() para construir progresivamente la lista de temperaturas convertidas evidencia comprensión operativa de estructuras dinámicas de datos. Esta técnica, más sofisticada que la asignación directa observada en niveles anteriores, refleja lo que Borba et al. (2016) identifican como desarrollo del pensamiento algorítmico-computacional integrado con el pensamiento matemático.

Estructura algorítmica jerárquica: El código muestra una estructura jerárquica donde primero se definen las colecciones de datos y luego se implementan transformaciones sobre ellas. Esta organización, según Schoenfeld (1992), evidencia el desarrollo del “conocimiento estratégico” y la capacidad de planificar y estructurar secuencias de operaciones para resolver problemas complejos.

Intención de cálculo de variaciones temporales: El inicio de un segundo bucle destinado a calcular variaciones (variaciones = [] y for i in:) revela comprensión conceptual de la necesidad de analizar cambios entre valores consecutivos en series temporales. Esta comprensión, aunque no implementada completamente, evidencia desarrollo del pensamiento variacional (MEN,1998).

Análisis de capacidades cognitivas subyacentes:

Pensamiento computacional integrado: A diferencia de los niveles anteriores, donde la programación funciona principalmente como medio de cálculo, este código muestra integración efectiva entre pensamiento matemático y computacional. Esta integración, según Wing (2006), se caracteriza por la capacidad de formular problemas de manera que puedan ser resueltos computacionalmente, organizando datos y diseñando algoritmos apropiados.

Procesamiento vectorizado del pensamiento: La conceptualización y manipulación de listas completas como unidades (en lugar de valores individuales) evidencia lo que podríamos denominar "vectorización conceptual" del pensamiento, donde las operaciones se conciben simultáneamente sobre colecciones enteras. Esta capacidad, según Hoyles et al. (2010), es característica del pensamiento matemático expresado en términos tecnológicos en contextos avanzados.

Metacognición estratégica: La estructuración deliberada del código en fases (definición de datos, transformación, análisis de variaciones) refleja desarrollo de capacidades metacognitivas para planificar y organizar procesos de resolución de problemas. Esta metacognición, según Schoenfeld (1992), es componente esencial del pensamiento matemático avanzado.

Comprensión funcional de transformaciones de datos: La implementación correcta de la fórmula de conversión de temperaturas demuestra comprensión operativa de funciones como mapeos sistemáticos entre diferentes dominios o representaciones. Esta comprensión, según el MEN (1998), es característica avanzada del Pensamiento Variacional.

Abstracción iterativa desarrollada: La implementación de un bucle 'for' para procesar sistemáticamente cada elemento de una colección evidencia desarrollo de la "abstracción iterativa", la capacidad para conceptualizar y manipular procesos repetitivos mediante estructuras de control algorítmicas. Esta capacidad, según Wing (2006), es componente del pensamiento computacional.

Limitaciones identificadas:

Estructura algorítmica incompleta: El segundo bucle queda incompleto ("for i in": seguido de 'pass'), lo que sugiere comprensión conceptual de la necesidad de calcular variaciones, pero dificultad para implementar el algoritmo específico. Esta limitación refleja lo que Mason, Burton

y Stacey (2010) identifican com“ "desarrollo en cur”o" del pensamiento matemático-computacional.

Ausencia de procesamiento estadístico: Aunque el código prepara adecuadamente los datos, no incluye implementación de análisis estadísticos (medias, desviaciones, correlaciones) que serían relevantes para el contexto del problema. Esta limitación sugiere o falta de prácticas estadísticas en la escuela o desarrollo incipiente del Pensamiento Aleatorio (MEN,1998).

Falta de visualización gráfica: A pesar de la naturaleza de los datos (series temporales climáticas), el código no incluye implementación de visualizaciones gráficas que facilitarían su interpretación. Esta ausencia limita el aprovechamiento de lo que Pierce et al. (2011) identifican como una de las principales ventajas de las herramientas tecnológicas en matemáticas.

El análisis de este código evidencia Pensamiento Variacional y Aleatorio significativamente más desarrollado que en los niveles educativos anteriores, con capacidad para conceptualizar y manipular relaciones sistemáticas entre variables que cambian conjuntamente en el tiempo. La integración entre pensamiento matemático y computacional es más efectiva, manifestándose en la implementación de estructuras algorítmicas funcionales que calculan resultados, y reflejan comprensión operativa de conceptos matemáticos avanzados. Esta sinergia entre ambos tipos de pensamiento permite a los estudiantes modelar fenómenos complejos, manipular series de datos y establecer transformaciones sistemáticas que serían difíciles de conceptualizar sin herramientas computacionales. Las limitaciones identificadas representan oportunidades específicas para intervención didáctica orientada al desarrollo del pensamiento matemático-computacional.

4.2.6. Pensamiento Numérico Aplicado a Contextos Económicos en Décimo Grado

El código siguiente, desarrollado para analizar costos en el contexto de la pesca en Quibdó, revela características del pensamiento numérico aplicado a contextos económicos:

Figura 18:

Código sobre modelación económica realizado por un estudiante de décimo grado (cálculo de costos pesqueros).

Code

```

1 #Start code here
2 costo_lancha = 8500000
3 costo_combustible_diario = 9000
4 precio_pescado_kg = 12000
5 dias_trabajo_mes = 25
6 costo_combustible_mensual = costo_combustible_diario * dias_trabajo_mes
7 print(costo_combustible_mensual)
8 meses_a_C3_B1o = 12
9 costo_mensual_lancha = costo_lancha % meses_a_C3_B1o
10 kilos_para_lancha = costo_mensual_lancha % precio_pescado_kg
11 print("")
12

```

Análisis exhaustivo de estructuras conceptuales y manifestaciones del pensamiento económico-matemático:

Modelización cuantitativa de estructura económica compleja: El código evidencia un intento de modelización matemática de una situación económica multidimensional, considerando diferentes categorías de costos (inversión inicial, costos operativos diarios), temporalidades diversas (diario, mensual, anual) y relaciones entre dominios heterogéneos (monetario, temporal, físico). Esta estructura, según Kilpatrick et al. (2001), refleja desarrollo avanzado de la "competencia estratégica", la capacidad para formular y representar problemas matemáticamente complejos.

Nomenclatura semánticamente estructurada: El patrón de nombrado de variables (costo_lancha, precio_pescado_kg, dias_trabajo_mes) revela organización cognitiva sistemática, donde cada nombre incorpora tanto la categoría conceptual (costo, precio, días) como el contexto específico (lancha, pescado, trabajo). Esta estructuración, según Schoenfeld (1992),

evidencia desarrollo de lo que denominamos "sistemas de conocimiento organizados", característicos del pensamiento matemático avanzado.

Escalonamiento temporal de variables económicas: La programación incluye variables para diferentes escalas temporales (diario, mensual, anual) y cálculos para convertir entre estas escalas ($\text{costo_combustible_mensual} = \text{costo_combustible_diario} \times 109 \text{ días_trabajo_mes}$). Esta estructura, según el MEN (1998), evidencia comprensión avanzada de la proporcionalidad y la variación en contextos temporales, aspectos del Pensamiento Variacional.

Amortización implícita de inversión de capital: El intento de calcular $\text{costo_mensual_lancha} = \text{costo_lancha} \% \text{ meses_a_C3_B10}$ (aunque con operador incorrecto) revela comprensión de la necesidad de distribuir el costo de inversión inicial (lancha) a lo largo de su vida útil. Este concepto, según Hoyles et al. (2010), representa un aspecto avanzado del pensamiento matemático aplicado a contextos económicos, donde se reconoce la necesidad de prorratear costos fijos en períodos operativos.

Conversión entre dominios económico y físico: El intento de calcular $\text{kilos_para_lancha} = \text{costo_mensual_lancha} \% \text{ precio_pescado_kg}$ (aunque con operador incorrecto) revela identificación de la necesidad de traducir valores monetarios (costo de la lancha) a unidades físicas equivalentes (kilogramos de pescado necesarios). Esta traducción, según Mason, Burton y Stacey (2010), evidencia desarrollo del pensamiento proporcional avanzado, donde se establecen equivalencias funcionales entre dominios heterogéneos.

Obstáculos epistemológicos significativos:

Confusión crítica entre operadores matemáticos: El uso del operador de módulo (%) en lugar del operador de división (/) para calcular el costo mensual del costo de operación de la lancha y los kilos equivalentes representa un obstáculo epistemológico. Este error va más allá de lo

meramente sintáctico y refleja lo que Brousseau (1997) denominaría u“ "obstáculo didácti”o" derivado de la transposición entre sistemas semióticos matemáticos y computacionales. La confusión revela dificultades en la traducción entre la operación matemática abstracta (división) y su implementación sintáctica específica en Python.

Modelo económico incompleto: El código implementa solo una parte del modelo económico necesario para responder completamente al problema planteado. No incluye cálculos de ingresos, ganancias o tiempos de recuperación de inversión, lo que limita su capacidad para resolver la situación económica completa. Esta limitación, refleja desarrollo parcial de l“ "competencia estratégi”a" para formular completamente problemas complejos (Kilpatrick et al., 2001).

Ausencia de estructuras condicionales y bucles: A pesar de la naturaleza del problema (que podría requerir análisis de diferentes escenarios), el código no implementa estructuras de control como condicionales o bucles que permitirían explorar diferentes situaciones. Esta ausencia, según Wing (2006), limita el aprovechamiento del potencial del pensamiento computacional para enriquecer el análisis matemático. Parece apenas natural que los estudiantes no incluyan condicionales o bucles, en tanto que tales prácticas no son usuales en las clases de matemáticas.

Interpretación limitada de resultados: El código muestra escaso desarrollo de la interpretación contextual de los resultados numéricos. Más allá de imprimir el valor del costo mensual de combustible, no incluye mensajes explicativos ni transformaciones adicionales que faciliten la comprensión del significado de los resultados en el contexto del problema. Esta limitación, según Schoenfeld (1992), evidencia un desarrollo incompleto de lo que denomina *conocimiento condicional*, es decir, la comprensión de ‘cuándo’ y ‘por qué’ determinados resultados son relevantes.

Análisis de patrones cognitivos subyacentes:

Matematización económica desarrollada: El código evidencia capacidad significativa para traducir una situación económica compleja a un modelo matemático estructurado. Esta traducción, según Freudenthal (1991), representa un aspecto avanzado de la "matematización horizontal", donde situaciones del mundo real se transforman en estructuras matemáticas formales.

Integración de múltiples temporalidades: La estructura desarrollada muestra comprensión de diferentes escalas temporales (diaria, mensual, anual) y su integración en un modelo coherente. Esta capacidad, según el MEN (1998), es característica del pensamiento variacional avanzado, donde se comprenden y manipulan diferentes tasas de cambio y periodicidades.

Jerarquización conceptual emergente: El código revela estructura jerárquica emergente, donde se distinguen diferentes categorías de costos (fijos/inversión vs. variables/operativos) y se establecen relaciones específicas entre ellos. Esta jerarquización, según Kilpatrick et al. (2001), es característica de la "comprensión conceptual" avanzada en matemáticas.

Pensamiento proporcional desarrollado: A pesar de los errores de implementación, el código evidencia comprensión de las relaciones proporcionales en economía: relaciones precio-cantidad, conversiones temporales (diario a mensual), y equivalencias entre diferentes tipos de valores (monetarios y físicos). Esta comprensión, según el MEN (1998), es característica avanzada del Pensamiento Numérico y Variacional.

Este código ejemplifica un nivel avanzado de integración entre pensamiento matemático y contextos económicos reales, con capacidad significativa para estructurar matemáticamente situaciones complejas, aunque con limitaciones importantes en la implementación técnica. Las estructuras cognitivas reveladas evidencian desarrollo significativo del pensamiento proporcional,

variacional y estratégico aplicado a contextos económicos, con obstáculos específicos en la traducción sintáctica entre el pensamiento matemático abstracto y su implementación computacional concreta.

4.2.7. Pensamiento Métrico y Geométrico en Estudiante de Sexto Grado

Abordando Tarea de Mayor Nivel

El siguiente código, desarrollado por un estudiante de sexto grado cuando abordaba una tarea originalmente diseñada para estudiantes de octavo grado sobre volúmenes y capacidades, revela características del pensamiento matemático:

Figura 19:

Código de estudiante de sexto grado aplicando la fórmula del volumen cilíndrico.

Code

```
1 #Start code here
2 diametro_tanque = 2
3 radio_tanque = diametro_tanque / 2
4 altura_tanque = 1.5
5 pi = 3.14159
6 radio_cuadrado = radio_tanque * radio_tanque
7 volumen_tanque_m3 = pi * radio_cuadrado * altura_tanque
8 print(volumen_tanque_m3)
9
```

Análisis de estructuras conceptuales y manifestaciones del pensamiento geométrico:

Comprensión avanzada de relaciones geométricas: El código revela una comprensión sofisticada, para un estudiante de sexto grado, de las relaciones entre diámetro y radio ($\text{radio_tanque} = \text{diámetro_tanque} / 2$), lo cual evidencia conocimiento de propiedades geométricas básicas del círculo. Este nivel de comprensión, según el MEN (1998), corresponde normalmente a etapas más avanzadas del desarrollo del Pensamiento Espacial y Geométrico.

Implementación algorítmica estructurada de fórmula volumétrica: El estudiante implementa la fórmula del volumen del cilindro ($\text{volumen_tanque_m}^3 = \pi * \text{radio_cuadrado} * \text{altura_tanque}$) de manera estructurada y secuencial, descompone correctamente el proceso en pasos lógicos: primero calcula el radio a partir del diámetro, luego el radio al cuadrado, y finalmente multiplica por π y por la altura. Esta descomposición algorítmica revela una capacidad de análisis procedimental generalmente asociada con estudiantes de mayor edad.

Aproximación precisa del valor de π : El uso de π con cinco decimales ($\pi = 3.14159$) en lugar de una aproximación más básica como 3.14, demuestra atención al detalle y comprensión de la importancia de la precisión en cálculos matemáticos, característica que según Kilpatrick et al. (2001) es un componente importante de la fluidez procedimental avanzada.

Uso correcto de variables intermedias: El estudiante crea una variable específica para calcular el radio al cuadrado ($\text{radio_cuadrado} = \text{radio_tanque} * \text{radio_tanque}$), lo que indica comprensión de modularidad en el cálculo y la importancia de descomponer expresiones complejas en componentes manejables. Esta práctica, según Mason, Burton y Stacey (2010), es característica de un pensamiento matemático que reconoce la importancia de la claridad procedimental.

Nomenclatura semánticamente significativa: El patrón de nombrado de variables es excepcionalmente claro y descriptivo (diámetro_tanque , radio_tanque , $\text{volumen_tanque_m}^3$), incorporando los conceptos matemáticos, el contexto físico y las unidades de medida. Esta práctica refleja una comprensión metacognitiva avanzada de la importancia de la claridad en la representación simbólica.

Análisis del nivel de desarrollo cognitivo-matemático:

Este código evidencia un caso notorio de desarrollo matemático avanzado para el nivel educativo del estudiante, mostrando:

Transferencia vertical de conocimientos: El estudiante evidencia capacidad para transferir conocimientos geométricos básicos (relación diámetro-radio) a contextos tridimensionales más complejos (cálculo de volumen cilíndrico), evidenciando lo que Tall (1981) denominó “crecimiento cognitivo vertical”, donde los conocimientos se extienden a dominios conceptuales más avanzados.

Integración conceptual-procedimental: A diferencia de otros casos analizados donde existe brecha entre comprensión conceptual e implementación procedimental, este estudiante muestra integración entre ambos aspectos, demostrando lo que Kilpatrick et al. (2001) describen como “entrecruzamiento” de hilos de competencia matemática.

Pensamiento algorítmico desarrollado: La implementación paso a paso de la fórmula, con variables intermedias claramente definidas, revela un pensamiento algorítmico estructurado que normalmente se desarrolla en etapas educativas posteriores. Esta estructuración, según Schoenfeld (1992), es característica de un pensamiento matemático que ha internalizado la importancia de la organización lógica en la resolución de problemas.

Abstracción geométrica tridimensional: La capacidad para aplicar y calcular el volumen de un objeto tridimensional partiendo de sus dimensiones básicas demuestra un nivel de abstracción geométrica que, según el modelo de Van Hiele (1986), correspondería al Nivel 2 (descriptivo/analítico) o incluso iniciando el Nivel 3 (abstracto/relacional), usualmente esperado en estudiantes de mayor edad.

Implicaciones pedagógicas específicas:

El caso de este estudiante de sexto grado que abordó exitosamente una tarea de octavo tiene importantes implicaciones didácticas:

Reconocimiento de desarrollo matemático heterogéneo: Confirma la necesidad de reconocer que el desarrollo del pensamiento matemático no siempre sigue patrones estrictamente asociados con la edad o nivel educativo formal, lo que requiere flexibilidad en la asignación de tareas según capacidades individuales.

Potencial de aceleración conceptual mediada tecnológicamente: Sugiere que entornos de programación bien estructurados (como el entorno de bloques visualizado en la figura 20) pueden funcionar como “aceleradores cognitivos”, permitiendo a estudiantes abordar conceptos normalmente reservados para niveles posteriores.

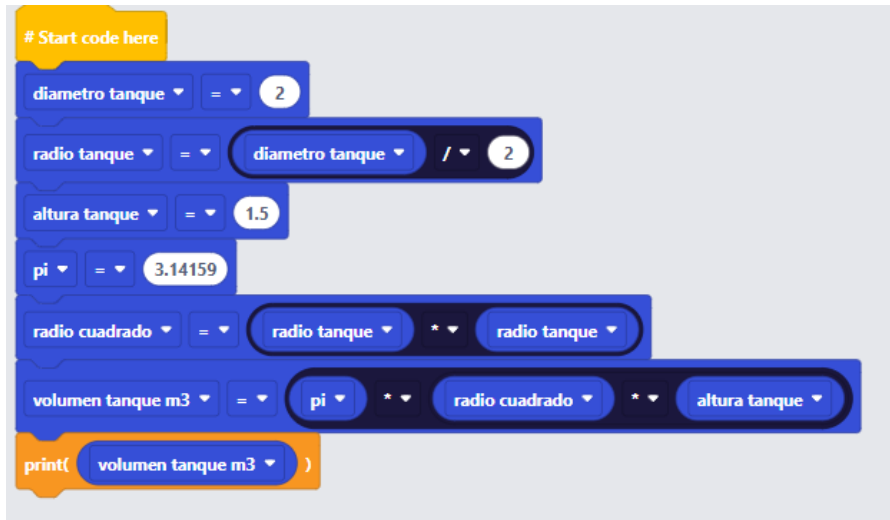
Valor de instrucciones claras y estructuradas: El éxito del estudiante “siguiendo las instrucciones” sugiere la importancia de proporcionar andamiajes cognitivos adecuados que permitan a los estudiantes acceder a conceptos más avanzados con el apoyo procedural necesario.

Oportunidades para diferenciación pedagógica: Casos como este justifican la implementación de estrategias de diferenciación que promuevan, a estudiantes con capacidades matemáticas avanzadas, abordar desafíos acordes con su nivel de desarrollo cognitivo, independientemente del grado educativo en el cual se ubica.

Este caso enriquece nuestra comprensión del pensamiento matemático manifestado por estudiantes cuando utilizan Python, lo que evidencia la existencia de trayectorias de desarrollo no lineales y el potencial de las herramientas computacionales para promover avances conceptuales más allá de los límites curriculares tradicionales.

Figura 20:

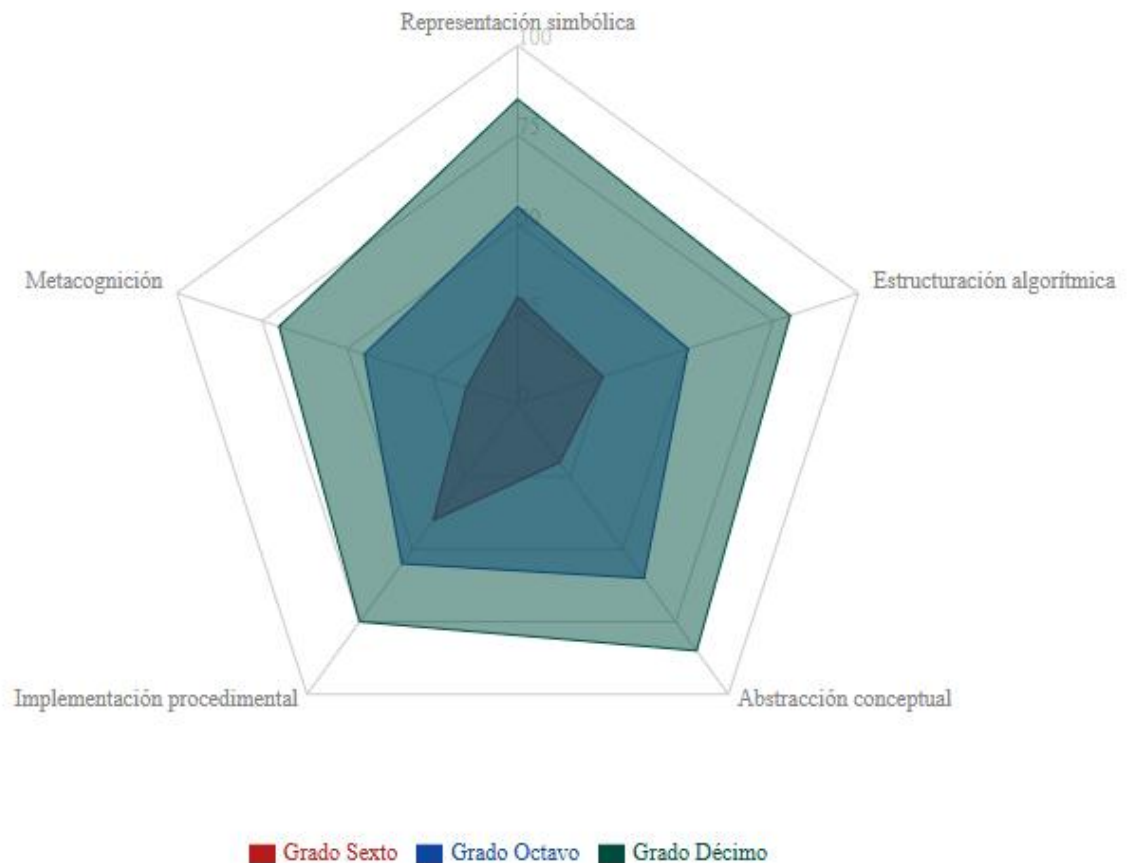
Representación en bloques (EduBlocks) del algoritmo para calcular el volumen del tanque, desarrollado por un estudiante de sexto grado.



El análisis transversal de los códigos Python desarrollados por los estudiantes revela patrones evolutivos consistentes en múltiples dimensiones del pensamiento matemático. La Figura 13 sintetiza esta evolución a través de cinco aspectos, evidenciando un desarrollo progresivo, pero no uniforme. Resulta el desarrollo acelerado de la abstracción conceptual entre octavo y décimo grado, así como el desarrollo comparativamente más lento de la dimensión metacognitiva a través de todos los niveles. Estos patrones sugieren oportunidades específicas para intervenciones didácticas orientadas a potenciar aspectos menos desarrollados del pensamiento matemático.

Figura 21:

Evolución del pensamiento matemático a través del código Python por nivel educativo



Nota: La gráfica muestra la evolución del pensamiento matemático en cinco dimensiones principales, observándose un desarrollo progresivo a medida que aumenta el nivel educativo. Los valores representan niveles de desarrollo en una escala de 0 a 100 basados en el análisis cualitativo de los códigos producidos por los estudiantes.

4.3. Triangulación analítica profunda de los patrones identificados

La triangulación sistemática de los análisis realizados sobre los códigos de estudiantes de diferentes niveles educativos permite identificar patrones evolutivos significativos en el pensamiento matemático manifestado a través de Python. Esta triangulación complementa y profundiza los análisis individuales, revelando tendencias, discontinuidades y transformaciones cualitativas en las estructuras cognitivas subyacentes.

4.3.1. Evolución estructural del pensamiento matemático a través de los niveles educativos

El análisis comparativo revela una transformación significativa en la estructura del pensamiento matemático a medida que se avanza en el nivel educativo:

1. En sexto grado predomina un pensamiento matemático que podríamos caracterizar como concreto-operacional específico, con las siguientes características distintivas:

Operatividad directa sobre valores específicos: Los algoritmos se construyen para trabajar directamente con los valores concretos del problema específico (tiempos de 4 y 6 horas, rectángulo de 3×2 metros).

Representación explícita de procesos intermedios: Cada paso del razonamiento tiende a representarse explícitamente (listas de múltiplos construidas manualmente, suma individual de lados para el perímetro).

Estructura secuencial lineal simple: Los algoritmos siguen secuencias lineales de operaciones discretas, con limitada jerarquización o agrupación conceptual.

Contextualización idiosincrática significativa: Fuerte tendencia a recontextualizar conceptos abstractos mediante referentes personales significativos (equipo_roblox).

Verificación implícita o ausente: Escasa implementación de mecanismos explícitos para verificar o validar resultados obtenidos.

Estos patrones se alinean con lo que Piaget e Inhelder (1969) caracterizaron como pensamiento operacional concreto, donde los estudiantes pueden realizar operaciones lógicas sobre objetos concretos, pero muestran limitaciones en la generalización y abstracción formal.

2. En octavo grado emerge un pensamiento que podríamos denominar conceptual-formalizado emergente, caracterizado por:

Formalización conceptual adelantada a la implementación: Mayor capacidad para formalizar y estructurar conceptos matemáticos, aunque con implementación procedural inconsistente o incompleta.

Desarrollo de meta-representaciones: Tendencia a utilizar el código para calcular conceptualmente estructuras matemáticas (RAIZ_CUADRADA_DE_SUMA__CUADRADOS).

Adopción de convenciones formales: Mayor atención a aspectos formales como el nombrado sistemático de variables y el uso de convenciones tipográficas (mayúsculas para constantes).

Estructuración jerárquica del conocimiento: Capacidad creciente para descomponer conceptos complejos en sus componentes (descomposición del Teorema de Pitágoras).

Discontinuidad procedural persistente: A pesar de la mayor formalización conceptual, permanecen discontinuidades significativas en la implementación procedural completa.

Estos patrones reflejan una etapa de transición cognitiva donde, según Tall y Vinner (1981), la imagen conceptual (comprensión personal del concepto) está en proceso de alineación con la definición conceptual formal (definición matemática aceptada), pero aún existen discrepancias significativas entre ambas.

3. En décimo grado se manifiesta un pensamiento relacional-sistémico integrado, con las siguientes características distintivas:

Estructuración multidimensional de datos: Capacidad para organizar y manipular estructuras de datos complejas y multidimensionales (series temporales paralelas, relaciones entre diferentes dominios).

Implementación algorítmica funcional: Mayor desarrollo de estructuras algorítmicas funcionales (bucles, métodos) que implementan operaciones sistemáticas sobre conjuntos de datos.

Modelización matemática contextualizada: Capacidad para traducir situaciones complejas del mundo real en modelos matemáticos estructurados (modelo económico de la pesca, análisis de datos climáticos).

Transferencia interdominios sistemática: Mayor sofisticación en la traducción entre diferentes dominios conceptuales (físico↔económico, Celsius↔Fahrenheit).

Integración cognitiva matemático-computacional: Mayor coherencia entre la conceptualización matemática y la implementación computacional, con aprovechamiento más efectivo de las estructuras algorítmicas.

Estos patrones se alinean con lo que Piaget e Inhelder (1958) describen como pensamiento formal: una lógica proposicional flexible y combinatoria, donde el adolescente puede razonar con múltiples enunciados (conjunciones, condicionales, disyunciones) y manejar variables simultáneamente en un dominio hipotético.

Esta evolución estructural del pensamiento matemático implica transformaciones cualitativas significativas en la organización cognitiva, la representación del conocimiento y las estrategias de resolución, más allá de cambios meramente cuantitativos. Refleja lo que Tall (1981) denomina "tres mundos matemáticos" (conceptual-encarnado, proceptual-simbólico y formal-axiomático), evidenciando cómo los estudiantes transitan progresivamente desde conceptualizaciones concretas hacia abstracciones formales y sistemáticas.

4.3.2. Patrones diferenciados por tipos de pensamiento matemático

La triangulación analítica revela patrones evolutivos diferenciados según los distintos tipos de pensamiento matemático establecidos por el MEN (1998):

Pensamiento numérico:

En sexto grado: Enfocado en operaciones directas sobre valores específicos, con comprensión concreta del MCM como intersección observable entre secuencias numéricas.

En octavo grado: Mayor reconocimiento de patrones numéricos significativos (terna pitagórica 3-4-5) y formalización conceptual de relaciones numéricas.

En décimo grado: Integración del pensamiento numérico en contextos aplicados complejos (análisis económico), con mayor capacidad para establecer relaciones proporcionales y funcionales entre valores.

Pensamiento espacial:

En sexto grado: Comprensión operativa de conceptos (área, perímetro) con implementación directa de fórmulas básicas, pero limitada comprensión dimensional.

En octavo grado: Formalización conceptual del Teorema de Pitágoras con comprensión de su estructura componencial, aunque con implementación incompleta.

En décimo grado: No se evidencia directamente en los códigos analizados, pero podría manifestarse en la capacidad para conceptualizar relaciones espaciales complejas en otros contextos.

Pensamiento métrico:

En sexto grado: Comprensión básica de unidades y conversiones simples ($\text{kg} \rightarrow \text{g}$), con manipulación directa de factores de conversión.

En octavo grado: Mayor capacidad para trabajar con medidas derivadas (áreas, volúmenes) y establecer relaciones entre diferentes tipos de magnitudes.

En décimo grado: Integración sofisticada de diferentes sistemas de medición (Celsius↔Fahrenheit) en contextos de análisis de datos reales.

Pensamiento variacional:

En sexto grado: Predominantemente implícito, con comprensión básica de proporcionalidad directa sin formalización explícita.

En octavo grado: Mayor reconocimiento de relaciones funcionales, aunque aún con implementación limitada o fragmentada.

En décimo grado: Altamente desarrollado, con capacidad para conceptualizar y manipular series temporales, transformaciones funcionales y relaciones sistemáticas entre variables.

Pensamiento aleatorio:

En sexto grado: Escasamente evidenciado en los códigos analizados, posiblemente reflejando menor desarrollo de este tipo de pensamiento en este nivel.

En octavo grado: Emergente en algunos contextos, aunque sin implementación algorítmica sustancial.

En décimo grado: Mayor desarrollo, evidenciado en la capacidad para organizar y analizar datos sistemáticamente, aunque con implementación estadística aún limitada.

Estos patrones diferenciados sugieren que los distintos tipos de pensamiento matemático no evolucionan uniformemente, sino que siguen trayectorias de desarrollo parcialmente independientes. Esta observación tiene implicaciones pedagógicas importantes, sugiriendo la necesidad de estrategias didácticas diferenciadas según el tipo específico de pensamiento matemático que se busque desarrollar.

4.3.3. Análisis de obstáculos epistemológicos persistentes y emergentes

La triangulación revela obstáculos epistemológicos significativos que persisten o emergen a través de los diferentes niveles educativos:

Obstáculos persistentes a través de todos los niveles:

Discontinuidad semiótica entre matemática y programación: Todos los niveles muestran, en diferentes grados, dificultades para traducir correctamente conceptos matemáticos a su implementación sintáctica específica en Python. Este obstáculo se manifiesta desde errores simples (confusión entre metros y metros cuadrados en sexto grado) hasta más complejos (uso de % en lugar de / para división en décimo grado).

Brecha entre comprensión conceptual e implementación procedimental: En todos los niveles se observa disparidad, aunque con reducción progresiva, entre la comprensión conceptual de estructuras matemáticas y la capacidad para implementarlas procedimentalmente en código funcional.

Verificación y validación limitadas: La escasa implementación de mecanismos explícitos para verificar la corrección de resultados o validar la pertinencia de soluciones representa un obstáculo persistente en todos los niveles.

Fragmentación en cadenas de transformaciones complejas: Todos los niveles muestran dificultades, aunque decrecientes, para implementar completamente cadenas de transformaciones que involucren múltiples pasos o conversiones entre diferentes dominios.

Obstáculos específicos de cada nivel educativo:

En sexto grado:

Concretización específica: Dificultad para generalizar más allá de los valores específicos del problema.

Confusión dimensional: Obstáculos para distinguir entre diferentes dimensiones (lineal vs. área) y sus unidades correspondientes.

Personalización excesiva: Tendencia a recontextualizar conceptos de manera tan idiosincrática que puede dificultar la comunicación matemática formal.

En octavo grado:

Formalismo sin funcionalidad: Tendencia a desarrollar representaciones formales que no logran implementarse funcionalmente.

Discontinuidad lógico-procedural: Dificultad para mantener coherencia lógica completa en la secuencia de operaciones.

Meta-representación sin operatividad: Uso de representaciones descriptivas (RAIZ_CUADRADA_DE_SUMA__CUADRADOS) sin correspondiente implementación operativa.

En décimo grado:

Confusión entre operadores semánticamente próximos: Dificultad para distinguir entre operadores con significados relacionados pero distintos (división vs. módulo).

Implementación parcial de modelos complejos: Tendencia a desarrollar modelos conceptualmente sofisticados, pero con implementación incompleta.

Subutilización de estructuras algorítmicas avanzadas: Limitado aprovechamiento de estructuras como condicionales o funciones que potenciarían el análisis matemático.

Estos obstáculos, de acuerdo con la teoría de Brousseau (1997) sobre obstáculos epistemológicos, no representan simplemente "errores" sino concepciones o formas de conocimiento que son funcionales en ciertos contextos, pero generan limitaciones en otros. Su

identificación sistemática proporciona información valiosa para el diseño de intervenciones didácticas específicas orientadas a facilitar su superación.

4.3.4. Análisis del impacto de Python como mediador semiótico

La triangulación permite identificar patrones significativos en cómo Python funciona como mediador semiótico en el desarrollo del pensamiento matemático:

Efectos potenciadores observados:

Externalización y objetivación del pensamiento: Python proporciona un medio para hacer visible y manipulable el pensamiento matemático, permitiendo a los estudiantes exteriorizar y objetivar sus conceptualizaciones. Esta externalización es especialmente evidente en la estructuración explícita de algoritmos como el cálculo del MCM o la implementación del Teorema de Pitágoras.

Desarrollo de meta-representaciones: La necesidad de nombrar variables y funciones fomenta el desarrollo de meta-representaciones, donde los estudiantes deben reflexionar sobre los conceptos matemáticos y cómo representarlos simbólicamente. Este proceso metacognitivo se evidencia en todos los niveles, con creciente sofisticación.

Facilitación de exploraciones sistemáticas: En niveles más avanzados, Python facilita exploraciones sistemáticas que serían tediosas manualmente, como la conversión de toda una serie temporal de temperaturas o el análisis de múltiples escenarios económicos.

Expresión de algoritmos como objetos de pensamiento: Python permite que los algoritmos matemáticos no sean solo procedimientos implícitos sino objetos explícitos de pensamiento que pueden ser analizados, modificados y optimizados.

Tensiones y discontinuidades identificadas:

Interferencia sintáctica en etapas iniciales: En los niveles iniciales, las particularidades sintácticas de Python pueden interferir con la expresión directa del pensamiento matemático, como se evidencia en errores de sintaxis o en implementaciones incompletas.

Conflicto entre notación matemática y programática: Todos los niveles muestran, en diferente grado, tensiones derivadas de diferencias entre la notación matemática tradicional y su implementación en Python (elevación al cuadrado, raíz cuadrada, módulo vs. división).

Sobrecarga cognitiva potencial: En algunos casos, especialmente en octavo grado, se observa que la necesidad de atender simultáneamente aspectos conceptuales matemáticos y requisitos sintácticos de programación puede generar sobrecarga cognitiva, resultando en implementaciones inconsistentes o incompletas.

Esta dualidad de Python tanto como impulsor como posible obstáculo en el desarrollo del pensamiento matemático se alinea con lo que Hegedus y Moreno-Armella (2009) describen como la naturaleza transformadora de la tecnología en educación matemática: no es una herramienta neutral sino un medio que reconfigura las formas de pensar y aprender matemáticas.

4.4. Caracterización integral del pensamiento matemático manifestado

Integrando los análisis individuales y la triangulación sistemática, es posible establecer una caracterización multidimensional del pensamiento matemático manifestado por los estudiantes al resolver tareas matemáticas mediante Python. Esta caracterización se estructura en cuatro dimensiones que capturan aspectos complementarios de la actividad matemática cognitiva.

4.4.1. Dimensión representacional-simbólica

El pensamiento matemático manifestado a través de Python se caracteriza por patrones específicos sobre la representación y manipulación simbólica de conceptos matemáticos:

Multimodalidad representacional evolutiva: A través de los diferentes niveles educativos, se observa evolución significativa en la capacidad para integrar y coordinar diferentes sistemas representacionales:

En sexto grado: Predominio de representaciones directas y específicas, con limitada conexión entre diferentes sistemas semióticos.

En octavo grado: Mayor capacidad para establecer conexiones entre representaciones verbales, numéricas y simbólicas, aunque con articulación incompleta.

En décimo grado: Integración más coherente de múltiples sistemas representacionales, incluyendo estructuras de datos computacionales como listas y variables estructuradas.

Hibridación semiótica progresiva: Se observa el desarrollo gradual de sistemas híbridos de representación que combinan elementos de la notación matemática tradicional con elementos de la sintaxis de programación:

En sexto grado: Yuxtaposición más que integración, con elementos de ambos sistemas coexistiendo sin fusión coherente.

En octavo grado: Mayor fusión conceptual, evidenciada en nomenclatura como `RAIZ_CUADRADA_DE_SUMA__CUADRADOS`, que combina expresividad matemática con estructura computacional.

En décimo grado: Integración más funcional, donde conceptos matemáticos se implementan a través de estructuras computacionales apropiadas ('for' para transformaciones sistemáticas).

Externalización cognitiva estructurada: Python funciona como medio de externalización del pensamiento matemático, permitiendo hacer explícitas estructuras conceptuales que normalmente permanecerían implícitas:

En sexto grado: Externalización principalmente de procesos de cálculo específicos y directos.

En octavo grado: Externalización de estructuras conceptuales más complejas, aunque con implementación incompleta.

En décimo grado: Externalización de modelos matemáticos estructurados y algoritmos de procesamiento sistemático.

Evolución en la abstracción simbólica: Se observa progresión significativa en el nivel de abstracción de las representaciones simbólicas utilizadas:

En sexto grado: Símbolos principalmente como representaciones directas de cantidades específicas.

En octavo grado: Símbolos como representaciones de conceptos y estructuras matemáticas generales.

En décimo grado: Símbolos como componentes de sistemas algorítmicos integrados, con mayor nivel de abstracción y generalidad.

Esta dimensión representacional-simbólica se alinea con lo que Duval (1995) identifica como aspecto del pensamiento matemático: la capacidad para coordinar diferentes registros semióticos y realizar transformaciones entre ellos. El análisis revela cómo Python puede funcionar como un sistema semiótico adicional que interactúa con los sistemas representacionales matemáticos tradicionales, generando tanto oportunidades de enriquecimiento como desafíos de integración.

4.4.2. Dimensión procedimental-estratégica

Esta dimensión captura cómo los estudiantes organizan, implementan y controlan procedimientos matemáticos a través de Python:

Evolución desde linealidad hacia estructuración algorítmica: Se observa una progresión desde procedimientos lineales simples hacia estructuras algorítmicas más complejas y organizadas:

En sexto grado: Predominio de secuencias lineales de operaciones discretas, con limitada estructuración algorítmica.

En octavo grado: Intentos de estructuración más compleja, aunque con implementación inconsistente o incompleta.

En décimo grado: Mayor desarrollo de estructuras algorítmicas funcionales, incluyendo bucles y métodos para procesamiento sistemático.

Fragmentación procedimental con integración progresiva: Todos los niveles muestran cierto grado de fragmentación en la implementación de procedimientos complejos, pero con integración creciente a medida que avanza el nivel educativo:

En sexto grado: Alta fragmentación, con operaciones discretas escasamente conectadas en flujos coherentes.

En octavo grado: Fragmentación moderada, con mayor intento de conexión lógica entre operaciones, aunque con discontinuidades persistentes.

En décimo grado: Menor fragmentación, con desarrollo de cadenas procedimentales más integradas y coherentes.

Verificación y validación emergentes pero limitadas: En todos los niveles se observa desarrollo limitado de estrategias explícitas para verificar resultados o validar procedimientos:

En sexto grado: Verificación prácticamente ausente, con aceptación directa de resultados calculados.

En octavo grado: Emergencia incipiente de conciencia sobre la necesidad de verificación, aunque sin implementación sistemática.

En décimo grado: Mayor desarrollo de la importancia de la verificación, aunque aún con implementación limitada en el código.

Evolución en la adaptabilidad estratégica: Se observa desarrollo progresivo de la capacidad para adaptar estrategias según las características específicas de cada problema:

En sexto grado: Estrategias rígidas y específicas para el problema concreto, con limitada transferibilidad.

En octavo grado: Mayor flexibilidad conceptual, aunque aún con dificultades para implementar estrategias adaptadas a nuevos contextos.

En décimo grado: Desarrollo más avanzado de estrategias adaptativas, con mayor capacidad para seleccionar enfoques apropiados según el contexto.

Esta dimensión procedimental-estratégica se relaciona directamente con lo que Schoenfeld (1992) identifica como componentes de la competencia matemática: conocimiento de base, estrategias de resolución de problemas, control y monitoreo, y sistemas de creencias. El análisis revela desarrollo significativo en conocimiento de base y estrategias, con menor evolución en aspectos de control y monitoreo, sugiriendo áreas específicas para intervención didáctica.

4.4.3. Dimensión conceptual-estructural

Esta dimensión se refiere a cómo los estudiantes comprenden, organizan y relacionan conceptos matemáticos a través de Python:

Reificación progresiva de conceptos matemáticos: Se observa evolución significativa en la capacidad para transformar procesos matemáticos en objetos conceptuales manipulables:

En sexto grado: Los conceptos se abordan principalmente como procesos operativos (por ejemplo, el área como una multiplicación y el perímetro como una suma

En octavo grado: Mayor reificación conceptual, aunque incompleta (Teorema de Pitágoras como estructura conceptual identificable).

En décimo grado: Desarrollo más avanzado de conceptos como objetos manipulables algorítmicamente (funciones de conversión, modelos económicos).

Estructuración jerárquica emergente: Se observa desarrollo progresivo de capacidades para organizar conceptos en estructuras jerárquicas coherentes:

En sexto grado: Limitada jerarquización con conceptos principalmente en un mismo nivel de organización.

En octavo grado: Emergencia de estructuración jerárquica, con descomposición de conceptos complejos en componentes más simples.

En décimo grado: Mayor desarrollo de estructuras conceptuales jerárquicas, con organización de modelos complejos en componentes interrelacionados.

Articulación conceptual evolutiva: Se observa progresión en la capacidad para establecer conexiones significativas entre diferentes conceptos matemáticos:

En sexto grado: Conexiones principalmente implícitas o superficiales entre conceptos relacionados.

En octavo grado: Mayor reconocimiento de conexiones conceptuales, aunque con articulación incompleta.

En décimo grado: Articulación más desarrollada entre conceptos de diferentes dominios y áreas matemáticas.

Contextualización conceptual diferenciada: Se observa evolución en cómo los estudiantes contextualizan conceptos matemáticos abstractos:

En sexto grado: Contextualización idiosincrática y personal (equipo_roblox), con fuerte anclaje en referentes familiares.

En octavo grado: Contextualización más alineada con aplicaciones matemáticas convencionales, aunque aún con elementos personales.

En décimo grado: Contextualización más sofisticada en modelos aplicados complejos (análisis económico, datos climáticos).

Esta dimensión conceptual-estructural se relaciona con lo que Kilpatrick et al. (2001) denominan "comprensión conceptual", la comprensión integrada y funcional de ideas matemáticas. El análisis revela desarrollo significativo en esta dimensión a través de los niveles educativos, con progresión desde comprensiones fragmentadas y específicas hacia estructuraciones más coherentes e integradas.

4.4.4. Dimensión metacognitiva-reflexiva

Esta dimensión captura aspectos relacionados con la conciencia, el control y la reflexión de los estudiantes sobre sus propios procesos de pensamiento matemático:

Monitoreo de coherencia global en desarrollo: Se observa evolución en la capacidad para evaluar la coherencia global de las soluciones desarrolladas:

En sexto grado: Foco principalmente en la funcionalidad local de operaciones específicas, con limitada atención a la coherencia global.

En octavo grado: Mayor conciencia de la importancia de la coherencia global, aunque con dificultades para mantenerla consistentemente.

En décimo grado: Desarrollo más avanzado del monitoreo de coherencia global, con mayor capacidad para mantener consistencia a través de procesos complejos.

Conciencia representacional evolutiva: Se observa desarrollo progresivo de conciencia sobre el impacto de diferentes formas de representación:

En sexto grado: Escasa evidencia de reflexión explícita sobre formas de representación, con selección principalmente intuitiva.

En octavo grado: Mayor conciencia sobre diferentes posibilidades representacionales, evidenciada en intentos de formalización tipográfica.

En décimo grado: Conciencia más desarrollada sobre la importancia de seleccionar representaciones apropiadas para diferentes contextos y propósitos.

Autorregulación estratégica diferenciada: Se observa evolución en la capacidad para adaptar estrategias según características específicas de cada problema:

En sexto grado: Autorregulación limitada, con aproximaciones principalmente directas y específicas.

En octavo grado: Mayor desarrollo de capacidades autorregulativas, aunque con implementación inconsistente.

En décimo grado: Autorregulación más avanzada, evidenciada en la selección de estructuras algorítmicas apropiadas para diferentes contextos.

Reflexión sobre resultados y procesos: Se observa desarrollo gradual en la capacidad para reflexionar sobre resultados y procesos matemáticos:

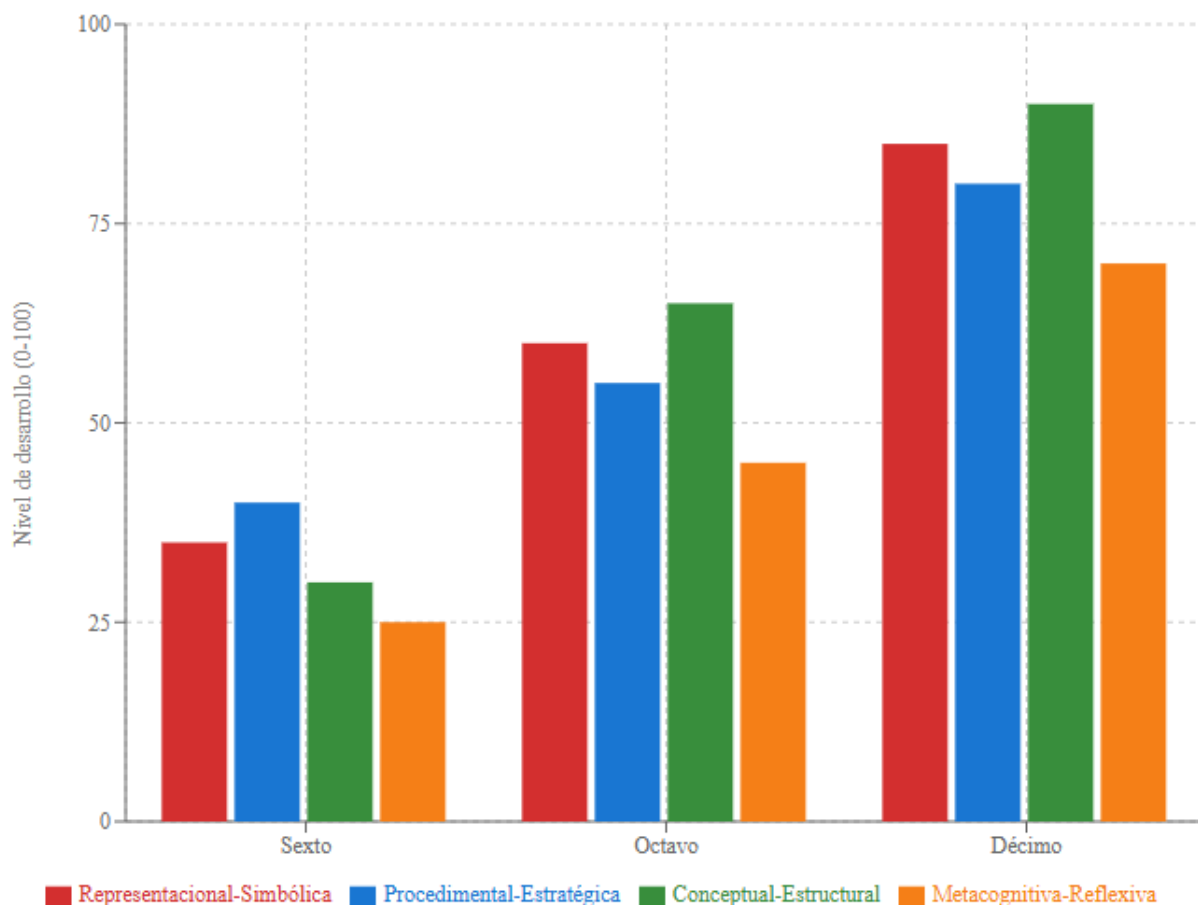
En sexto grado: Limitada evidencia de reflexión explícita, con foco principalmente en resultados numéricos directos.

En octavo grado: Mayor reflexión conceptual, aunque con limitada implementación de esta reflexión en el proceso de solución.

En décimo grado: Reflexión más desarrollada, manifestada en la interpretación contextualizada de resultados y en la selección de enfoques apropiados.

Esta dimensión metacognitiva-reflexiva se alinea con lo que Schoenfeld (1992) identifica como aspectos del pensamiento matemático avanzado: control, monitoreo y autorregulación de los procesos cognitivos. El análisis revela desarrollo moderado en esta dimensión, con oportunidades significativas para potenciar estos aspectos metacognitivos a través de intervenciones didácticas específicas.

La caracterización integral del pensamiento matemático manifestado por los estudiantes puede organizarse en cuatro dimensiones que capturan aspectos complementarios de la actividad matemática cognitiva. La Figura 14 compara el desarrollo de estas dimensiones a través de los tres grados analizados, evidenciando que la dimensión conceptual-estructural alcanza los niveles más altos de desarrollo en décimo grado, mientras que la dimensión metacognitiva-reflexiva muestra el desarrollo comparativamente más limitado en todos los niveles. Esta diferenciación dimensional proporciona información valiosa para el diseño de intervenciones didácticas específicas que atiendan de manera selectiva las diferentes facetas del pensamiento matemático.

Figura 22:*Comparativa de las cuatro dimensiones del pensamiento matemático por grado*

Nota: La gráfica muestra el nivel de desarrollo de las cuatro dimensiones es del pensamiento matemático identificadas en el estudio, comparando su evolución a través de los tres grados analizados. Se observa un desarrollo consistente en todas las dimensiones con el avance del nivel educativo, aunque con diferente intensidad.

4.5. Implicaciones didácticas derivadas del análisis

El análisis exhaustivo realizado permite identificar implicaciones didácticas para el diseño de intervenciones orientadas al desarrollo del pensamiento matemático mediante Python:

4.5.1. Estrategias para abordar conflictos semióticos identificados

Los conflictos semióticos identificados, especialmente aquellos relacionados con la traducción entre diferentes sistemas de representación, requieren intervenciones específicas:

Desarrollo de puentes semióticos explícitos: Es importante diseñar actividades que establezcan conexiones explícitas entre diferentes sistemas de representación, particularmente entre notación matemática tradicional y su implementación en Python. Estas actividades podrían incluir:

- Tablas de equivalencia entre operaciones matemáticas y su sintaxis en Python
- Ejercicios de traducción bidireccional entre notación matemática y código
- Análisis comparativo de la misma solución expresada en diferentes sistemas representacionales

Secuenciación estratégica en la introducción de sintaxis: El análisis sugiere la importancia de introducir gradualmente la sintaxis de Python, partiendo de estructuras más cercanas a la notación matemática tradicional y avanzando progresivamente hacia construcciones más específicamente computacionales. Esta secuenciación podría seguir el patrón:

- Operaciones aritméticas básicas y variables simples
- Estructuras de datos simples (listas) y operaciones sobre ellas
- Estructuras de control (condicionales, bucles) con progresiva complejidad
- Funciones y modularización de código

Estrategias de implementación paralela: Para abordar la brecha entre comprensión conceptual e implementación procedimental, resulta valioso fomentar implementaciones paralelas del mismo concepto en diferentes representaciones:

- Implementación en notación matemática tradicional
- Implementación en pseudocódigo intermedio
- Implementación en Python completo
- Análisis comparativo de las tres representaciones

Estas estrategias se alinean con lo propuesto por Duval (1995) sobre la importancia de desarrollar la capacidad para navegar fluidamente entre diferentes registros semióticos como componente del pensamiento matemático.

4.5.2. Estrategias para potenciar el desarrollo del pensamiento matemático según nivel educativo

El análisis revela necesidades didácticas diferenciadas según el nivel educativo:

Para estudiantes de sexto grado:

Fomento de generalización progresiva: Diseñar actividades que partan de ejemplos específicos, pero estimulen la progresiva generalización algorítmica.

Desarrollo de estructuras de verificación: Introducir sistemáticamente prácticas de verificación y validación de resultados como componente de toda solución.

Conexión explícita entre contextos y conceptos: Aprovechar la tendencia a contextualizar idiosincráticamente los conceptos, pero estableciendo conexiones explícitas con formulaciones matemáticas formales.

Clarificación dimensional: Abordar específicamente la distinción entre diferentes dimensiones (lineal, área, volumen) y sus unidades correspondientes.

Para estudiantes de octavo grado:

Integración concepto-procedimiento: Diseñar actividades que aborden explícitamente la brecha entre comprensión conceptual y capacidad de implementación procedimental.

Operativización de meta representaciones: Fomentar la transformación de representaciones conceptuales (RAIZ_CUADRADA_DE_SUMA__CUADRADOS) en implementaciones operativas funcionales.

Desarrollo de coherencia lógico-algorítmica: Trabajar específicamente en el desarrollo de secuencias coherentes y completas de operaciones para implementar conceptos complejos.

Formalización progresiva: Aprovechar la tendencia a la formalización, orientándola hacia convenciones estándar y efectivas.

Para estudiantes de décimo grado:

Refinamiento sintáctico específico: Abordar específicamente confusiones entre operadores semánticamente próximos (división/módulo) mediante análisis comparativo explícito.

Implementación de estructuras algorítmicas avanzadas: Fomentar el uso de estructuras como condicionales, bucles anidados y funciones para potenciar análisis matemáticos complejos.

Desarrollo de verificación sistemática: Incorporar prácticas sistemáticas de verificación, validación y análisis de resultados como componente de soluciones avanzadas.

Integración interdisciplinaria: Promover conexiones explícitas entre diferentes dominios (matemático, computacional, contextual) en la resolución de problemas complejos.

Estas estrategias diferenciadas se alinean con la visión de Vygotsky (1978) sobre la importancia de trabajar en la "zona de desarrollo próximo" de cada estudiante, ofreciendo desafíos apropiados a su nivel actual de desarrollo pero que estimulen su progresión hacia estructuras cognitivas más avanzadas.

4.5.3. Aprovechamiento didáctico de Python como mediador semiótico

El análisis revela oportunidades específicas para aprovechar didácticamente el potencial de Python como mediador semiótico en el desarrollo del pensamiento matemático:

Potenciación de la visualización dinámicas: Aprovechar las capacidades gráficas de Python para facilitar la exploración y comprensión de conceptos matemáticos complejos. Esto puede incluir:

- Representación visual de comportamientos funcionales
- Visualización de transformaciones geométricas
- Exploración interactiva de modelos matemáticos

Implementación de exploración paramétrica sistemática: Utilizar Python para implementar exploraciones sistemáticas de cómo diferentes parámetros afectan comportamientos matemáticos, facilitando:

- Descubrimiento de patrones y regularidades
- Comprensión de efectos de cambios en variables
- Formulación y verificación de conjeturas

Desarrollo de modelos incrementales: Fomentar la implementación de aproximaciones progresivamente más sofisticadas a un mismo problema matemático:

- Implementación inicial con valores específicos
- Generalización a fórmulas paramétricas
- Desarrollo de algoritmos adaptativos
- Optimización y refinamiento de soluciones

Integración de verificación algorítmica: Incorporar componentes de verificación algorítmica en las soluciones desarrolladas:

- Verificación automática de resultados
- Comprobación de condiciones de validez
- Análisis de casos especiales o extremos

Estas estrategias se alinean con lo propuesto por Hoyles et al. (2010) sobre cómo las tecnologías digitales pueden transformar las prácticas matemáticas, facilitando cálculos y

reconfigurando las formas en que se exploran, representan y comprenden los conceptos matemáticos.

4.5.4. Desarrollo intencional de capacidades metacognitivas

El análisis revela la necesidad de estrategias específicas para potenciar el desarrollo metacognitivo, un aspecto con desarrollo limitado en todos los niveles:

Incorporación explícita de documentación reflexiva: Promover prácticas sistemáticas de documentación de código que incluyan:

- Explicación del razonamiento matemático subyacente
- Justificación de decisiones algorítmicas tomadas
- Análisis de fortalezas y limitaciones de la solución implementada
- Identificación de posibles mejoras o alternativas

Implementación de actividades de depuración analítica: Utilizar actividades específicas de identificación y corrección de errores como oportunidades para desarrollar capacidades metacognitivas:

- Análisis de código con errores conceptuales o procedimentales
- Identificación de inconsistencias lógicas o matemáticas
- Implementación de correcciones justificadas
- Reflexión sobre patrones comunes de error

Promoción de análisis comparativo de soluciones: Fomentar la evaluación explícita de diferentes aproximaciones a un mismo problema:

- Comparación de eficiencia algorítmica
- Análisis de claridad y legibilidad de código
- Evaluación de generalidad y adaptabilidad

- Identificación de ventajas contextuales específicas

Desarrollo de prácticas de transformación representacional explícitas: Implementar actividades que requieran transformación explícita entre diferentes sistemas representacionales:

- Traducción de problema verbal a modelo matemático
- Transformación de modelo matemático a algoritmo
- Conversión de algoritmo a código Python
- Reflexión sobre cambios y adaptaciones necesarias en cada transformación

Estas estrategias metacognitivas se alinean con lo propuesto por Schoenfeld (1992) sobre la importancia crucial del desarrollo de capacidades de control y monitoreo para el pensamiento matemático avanzado, y ofrecen oportunidades específicas para potenciar estos aspectos a través de la programación en Python.

4.6. Visualización de patrones identificados

Code

```
1 #Start code here
2 tiempo_lancha1 = 4
3 tiempo_lancha2 = 6
4 hora_inicial = 6
5 multiplo_lancha1 = [4, 8, 12, 16, 20, 24]
6 multiplo_lancha2 = [6, 12, 18, 24]
7 mcm = 12
8 proxima_concidencia = hora_inicial + mcm
9 print("6:00pm")
10
```

Code

```

1 #Start code here
2 BASE_POSTE = 4
3 ALTURA_POSTE = 3
4 SUMA__CUADRADOS = BASE_CUADRADO + ALTURA_CUADRADO
5 LONGITUD__POSTE = RAIZ_CUADRADA_DE_SUMA__CUADRADOS
6 LADO_1 = 3
7 LADO_2 = 4
8 LADO_3 = 5
9 LADO_1__CUADRADO = BASE_POSTE
10 ALTURA_CUADRADO = ALTURA_POSTE * ALTURA_POSTE
11

```

Code

```

1 #Start code here
2 meses = ["enero", "febrero", "marzo", "abril", "mayo", "junio"]
3 temperatura_C_22 = [28, 29, 27, 30, 29, 28]
4 precipitaciones = [650, 700, 800, 750, 600, 550]
5 temperatura_F_22 = []
6 for temp in temperatura_C_22:
7     temp_F_22 = temp * 9/5 + 32
8     temperatura_F_22.append(temp_F_22)
9 variaciones = []
10 for i in :
11     pass
12

```

La presente caracterización multidimensional del pensamiento matemático manifestado por los estudiantes al utilizar Python como herramienta para la resolución de tareas proporciona fundamentos sólidos para el diseño de intervenciones didácticas específicas. Los hallazgos sugieren que la integración adecuada de Python en la enseñanza de las matemáticas puede potenciar el desarrollo del pensamiento matemático, siempre que se atiendan los obstáculos epistemológicos identificados y se aprovechen las oportunidades de representación, exploración y modelización que ofrece esta herramienta.

El análisis revela que Python no es meramente un instrumento auxiliar para el cálculo matemático, sino un mediador semiótico que transforma cualitativamente los procesos de pensamiento, generando nuevas posibilidades cognitivas, pero también desafíos específicos que

requieren intervención didáctica intencional. La consideración cuidadosa de estos aspectos puede contribuir significativamente a la efectividad de propuestas pedagógicas que integren programación y matemáticas, enriqueciendo el desarrollo del pensamiento matemático en contextos educativos diversos.

4.7. Análisis de las entrevistas finales

4.7.1. Metodología y estructura del análisis

Las entrevistas se estructuraron en torno a cinco preguntas:

- ¿Qué fue lo más interesante de resolver tareas matemáticas con Python?
- ¿Te resultó más fácil o difícil resolver tareas usando Python? ¿Por qué?
- ¿Qué aprendiste sobre matemáticas que no sabías antes?
- ¿En qué momento te sentiste más confundido? ¿Cómo lo resolviste?
- ¿Prefieres resolver problemas con papel y lápiz o con Python? ¿Por qué?

Se analizaron las respuestas utilizando un enfoque mixto que combina análisis cualitativo temático con representación cuantitativa de tendencias.

4.7.2. Resultados por Pregunta

Aspectos de mayor interés al resolver problemas matemáticos con Python

Tabla 7.

Categorización de aspectos de mayor interés

Categoría	Descripción	Frecuencia	Porcentaje	Ejemplos de respuestas
Facilidad y automatización	Referencias a la facilidad para resolver operaciones de manera automatizada	5	29.4%	"Ver como resuelve las operaciones con mas facilidad" (E1), "Como los resultados se resolvían solos" (E4)
Nuevas formas de resolución	Mención al descubrimiento de metodologías alternativas	4	23.5%	"Que descubri nuevas formas de resolver problemas matemáticos" (E1), "Aprender a usar este método el cual no conocía" (E2)
Aplicación contextualizada	Referencias a aplicaciones prácticas específicas	3	17.6%	"Fue calcular el costo de alimentos" (E3), "Lo que más me intereso fue ordenar los precios de cada uno costos" (E5)
Precisión y exactitud	Valoración de la exactitud en los resultados	2	11.8%	"Que nos dio el resultado exacto sin error y sin dificultades" (E6)
Trabajo colaborativo	Referencias al valor del trabajo en equipo	2	11.8%	"Que siempre trabajamos en grupo y cada uno daba su aporte" (E11)
Otra/No específica	Respuestas generales o no categorizables	1	5.9%	"Bien aprendí mucho" (E13)

Nota: La tabla muestra los aspectos que los estudiantes identificaron como más interesantes al resolver problemas matemáticos con Python, destacando la facilidad para resolver operaciones y el descubrimiento de nuevas metodologías.

La distribución de respuestas muestra que los aspectos más valorados por los estudiantes fueron la facilidad y automatización en la resolución de operaciones (29.4%) y el descubrimiento de nuevas formas de resolución de problemas (23.5%). Esto sugiere que Python logró captar el

interés de los estudiantes principalmente a través de su capacidad para simplificar procesos y ofrecer métodos alternativos de resolución, lo cual se alinea con lo planteado por Altamirano et al. (2023) sobre cómo Python facilita la realización de cálculos matemáticos.

Nivel de dificultad percibido al resolver problemas con Python

Tabla 8.

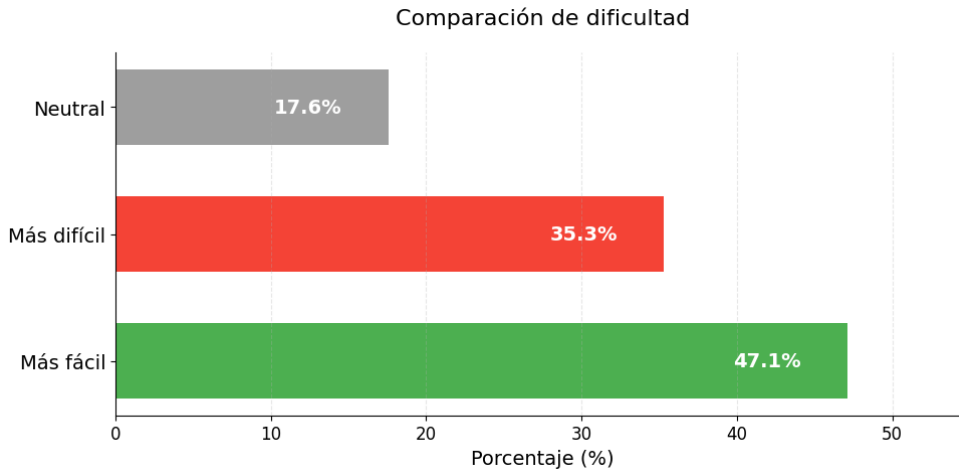
Percepción de dificultad en el uso de Python

Percepción	Frecuencia	Porcentaje	Razones principales	Ejemplos de respuestas
Más fácil	8	47.1%	- Automatización - Rapidez - Trabajo colaborativo	"Más fácil porque es poner los datos con sus respectivas funciones y agregar el cálculo" (E7)
				Fácil porque descubrí cosas nuevas" (E13)
Más difícil	6	35.3%	Problemas técnicos	"Más difícil ya que los computadores no funcionan muy bien y dificultan el trabajo" (E1)
			Dificultad con variables	
			Falta de familiaridad	"Muy difícil porque no es fácil aprender a usar las variables" (E1)
Neutral/Ambiguo	3	17.6%	Balance entre facilidad y complejidad	"Porque no fue tan fácil ni difícil todo tuve que resolver los costos solo me pareció un poco difícil en los ordenamientos" (E5)
			Aspectos específicos de dificultad	

Nota: La tabla categoriza las percepciones de los estudiantes respecto a la dificultad en el uso de Python, evidenciando una división entre quienes encontraron la herramienta facilitadora y quienes experimentaron mayor desafío.

Figura 23:

Distribución de percepciones sobre dificultad en el uso de Python



Nota: La gráfica muestra la distribución de percepciones de los estudiantes respecto a la dificultad experimentada al utilizar Python como herramienta para resolver problemas matemáticos, evidenciando una división significativa entre facilidad percibida (47.1%) y dificultad reportada (35.3%).

El análisis muestra que casi la mitad de los estudiantes (47.1%) encontraron que Python facilitaba la resolución de problemas matemáticos, mientras que un porcentaje significativo (35.3%) experimentó mayor dificultad. Esta división refleja lo que Weintrop et al. (2016) señalan sobre los desafíos iniciales en la adopción de herramientas computacionales para el aprendizaje matemático, pero también indica el potencial de Python como herramienta facilitadora cuando se superan estas barreras iniciales.

Entre las dificultades específicas mencionadas, destacan:

Problemas con la infraestructura tecnológica: "los computadores no funcionan muy bien y dificultan el trabajo" (E1)

Dificultad con el concepto de variables: "no es fácil aprender a usar las variables" (E1)

Pérdida de trabajo: "los computadores eliminan el trabajo que ya hiciste y te toca empezar de nuevo" (E1)

Entre los factores que facilitaron el trabajo, se mencionan:

1. **Automatización de cálculos:** "se resuelven solos" (E4)
2. **Trabajo colaborativo:** "me resulto más fácil por trabajar en grupo" (E11)
3. **Precisión y velocidad:** "ayuda a resolver problemas matemáticos con más sencillas, rápidos y fácil" (E3)

Aprendizajes matemáticos adquiridos mediante el uso de Python

Tabla 9.

Categorización de aprendizajes matemáticos reportados

Categoría de aprendizaje	Frecuencia	Porcentaje	Ejemplos de respuestas
Variables y conceptos programáticos	4	23.5%	"Las variables" (E4), "A al ser variables y muchas cosas más" (E17)
Resolución de problemas complejos	3	17.6%	"La resolución de problemas más largos" (E1), "Lo que aprendimos fue a multiplicar con números demasiados grandes" (E14)
Organización y planificación de datos	2	11.8%	"Planificar datos" (E3)
Aplicación interdisciplinaria	2	11.8%	"No sabíamos que la matemática se utiliza en informática" (E12), "A resolverlo en problemas o programas de tecnología" (E13)
Ningún aprendizaje nuevo	4	23.5%	"Nada nuevo porque ya todo lo sabía" (E1), "Todo lo que esta hay lo sabía" (E2)
Respuesta no específica	2	11.8%	"Que la matemática no es de solo número, si no que se usa en toda aria" (E15)

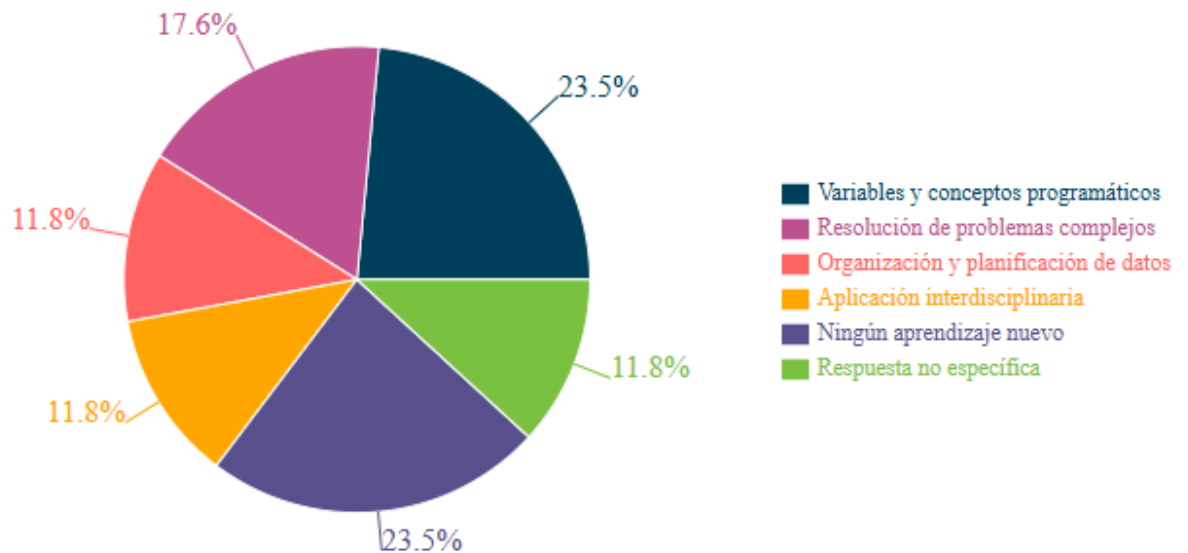
Nota: La tabla presenta los tipos de aprendizajes matemáticos que los estudiantes reportaron haber adquirido mediante el uso de Python, mostrando una polarización entre aprendizajes relacionados con variables y la percepción de no haber adquirido nuevos conocimientos.

La Figura 16 ilustra gráficamente la distribución de las categorías de aprendizajes matemáticos reportados por los estudiantes. Esta visualización permite apreciar claramente la polarización entre quienes identificaron aprendizajes relacionados con 'variables y conceptos programáticos' (23.5%) y aquellos que manifestaron no haber adquirido 'ningún aprendizaje nuevo'

(23.5%). Esta dualidad sugiere una bifurcación significativa en la percepción del valor educativo de Python, donde un grupo reconoce su contribución al desarrollo de conceptos algebraicos mientras otro no logra establecer conexiones entre programación y aprendizaje matemático.

Figura 24:

Distribución porcentual de las categorías de aprendizajes matemáticos reportados por los estudiantes



Nota: La gráfica ilustra la distribución de las categorías de aprendizajes matemáticos reportados, evidenciando una polarización entre quienes identificaron aprendizajes relacionados con variables y conceptos programáticos (23.5%) y aquellos que manifestaron no haber adquirido ningún aprendizaje nuevo (23.5%).

Es significativo que el mismo porcentaje de estudiantes (23.5%) reportó aprendizajes relacionados con variables y conceptos programáticos, como aquellos que afirmaron no haber adquirido ningún conocimiento matemático nuevo. Esta polarización sugiere una diferencia importante en cómo los estudiantes perciben la relación entre la programación y el aprendizaje matemático.

El aprendizaje sobre variables (23.5%) coincide con lo planteado por Supelano Mesa (2023) sobre cómo Python facilita la transición de procesos aritméticos a algebraicos. La mención a la resolución de problemas más complejos (17.6%) se alinea con los hallazgos de Arotuma y

Soraya (2017) sobre cómo la programación contribuye a desarrollar capacidades para resolver problemas más sofisticados.

Momentos de confusión y estrategias de resolución

Tabla 10.

Fuentes de confusión y estrategias de resolución

Fuente de confusión	Frecuencia	Porcentaje	Estrategia de resolución	Ejemplos
Manejo de variables	5	29.4%	Solicitar ayuda docente	"Cuando no sabía realizar bien las variables tuve que llamar al profesor para que me explicara" (E1), "Cuando puso esas vainas para resolver en el tablero" (E2)
Primeras interacciones con la plataforma	3	17.6%	Práctica y familiarización	"Cuando estábamos iniciando, cuando no había cogido bien porque no sabía usarla" (E8)
Aspectos específicos del problema	2	11.8%	Análisis y razonamiento	"Me sentí confundida cuando tuve que resolver las operaciones en una resolví analizando" (E3)
Sin confusión reportada	4	23.5%	N/A	"En ninguno" (E12)
Sin respuesta	3	17.6%	N/A	[Sin respuesta]

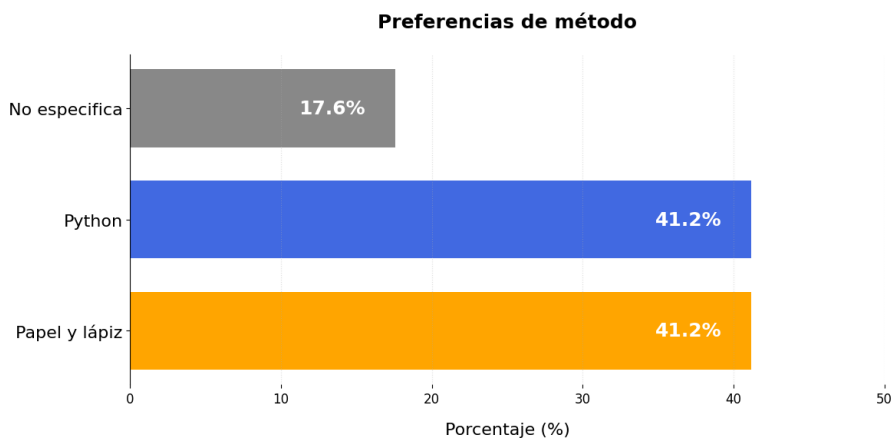
Nota: La tabla identifica las principales fuentes de confusión experimentadas por los estudiantes durante el proceso y las estrategias empleadas para resolverlas, destacando la importancia del apoyo docente.

El análisis muestra que la principal fuente de confusión fue el manejo de variables (29.4%), aspecto que coincide con las dificultades señaladas por Du Boulay (1980) respecto a los desafíos iniciales en la asimilación de conceptos programáticos. La estrategia predominante para resolver estas confusiones fue la solicitud de ayuda docente, lo que refuerza la importancia del rol del profesor como mediador en la introducción de herramientas tecnológicas.

Preferencias metodológicas: Python vs. papel y lápiz**Tabla 11.***Preferencias metodológicas y sus justificaciones*

Preferencia	Frecuencia	Porcentaje	Justificaciones principales	Ejemplos
Papel y lápiz	7	41.2%	Mayor familiaridad Mejor aprendizaje Más control del proceso	"Con papel y lápiz ya que se dificulta mucho en Python porque los computadores eliminan el trabajo que ya hiciste" (E1) "Con papel y lápiz lo encuentro más fácil porque de esa manera puedo aprender más" (E2)
Python	7	41.2%	Mayor facilidad Rapidez Precisión	"Con Python porque es más fácil resolver las operaciones" (E10) "Con Python porque es más precisa y creativo" (E3)
No especificado	3	17.6%	N/A	[Sin respuesta clara]

Nota: La tabla muestra la distribución equilibrada entre preferencias por Python y métodos tradicionales, junto con las justificaciones que fundamentan cada elección.

Figura 25:*Distribución de preferencias metodológicas*

Nota: La gráfica muestra la distribución perfectamente equilibrada (41.2% para cada método) entre las preferencias por Python y por papel y lápiz, revelando una división en las preferencias de aprendizaje de los estudiantes.

La distribución equilibrada (41.2% para cada método) entre las preferencias por Python y por papel y lápiz resulta interesante. Esta polarización refleja una división en las preferencias de aprendizaje de los estudiantes que debe ser considerada al implementar estrategias educativas tecnológicas.

Las justificaciones para preferir papel y lápiz se centran en:

Mayor control sobre el proceso: "Con papel y lápiz ya que se dificulta mucho en Python porque los computadores eliminan el trabajo que ya hiciste" (E1)

Percepción de mejor aprendizaje: "En papel ya que sentimos que aprendemos más a analizar y a comprender problemas matemáticos" (E17)

Familiaridad con el método: "Prefiero por ahora papel y lápiz porque no me desenvuelvo muy bien en la página" (E1)

Las justificaciones para preferir Python se enfocan en:

Facilidad y eficiencia: "Con Python porque es más fácil resolver las operaciones" (E10)

Precisión y creatividad: "Con Python porque es más precisa y creativo" (E3)

Ayuda en la resolución: "Me gusto más con Python porque se me facilito muchas cosas" (E8)

4.7.3. Análisis de Correlaciones entre Respuestas

4.7.3.1. Relación entre percepción de dificultad y preferencia metodológica

Tabla 12.

Correlación entre dificultad percibida y preferencia metodológica

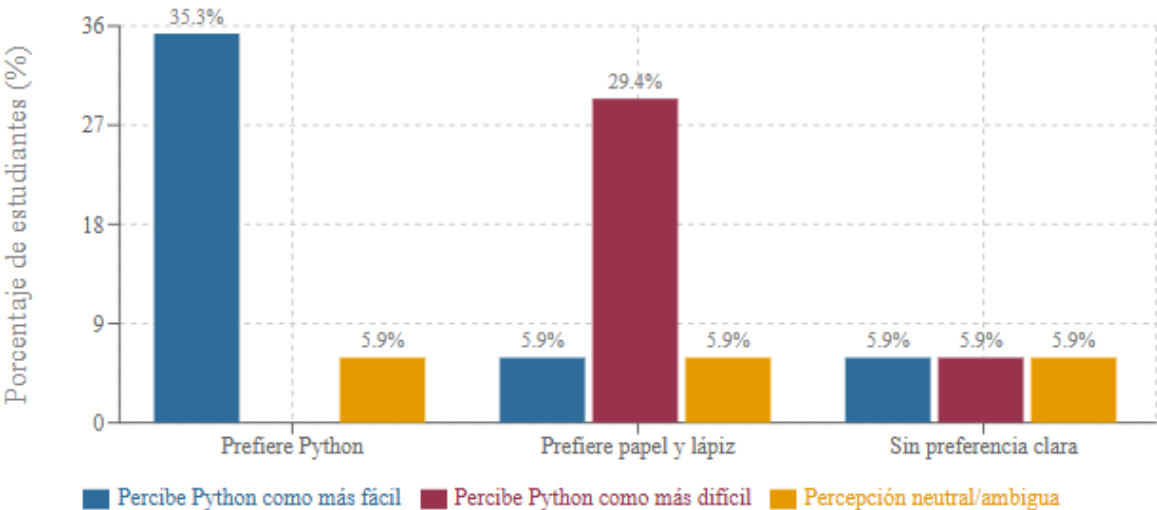
	Prefiere Python	Prefiere papel y lápiz	Sin preferencia clara
Percibe Python como más fácil	6 (35.3%)	1 (5.9%)	1 (5.9%)
Percibe Python como más difícil	0 (0%)	5 (29.4%)	1 (5.9%)
Percepción neutral/ambigua	1 (5.9%)	1 (5.9%)	1 (5.9%)

Nota: La tabla evidencia la fuerte correlación entre la percepción de dificultad en el uso de Python y la preferencia metodológica expresada por los estudiantes, revelando cómo la experiencia inicial influye en la aceptación de la herramienta.

La Figura 18 ilustra gráficamente la correlación entre la percepción de dificultad en el uso de Python y la preferencia metodológica expresada por los estudiantes.

Figura 26:

Correlación entre percepción de dificultad y preferencia metodológica



Nota: La gráfica ilustra la correlación entre la percepción de dificultad en el uso de Python y la preferencia metodológica expresada por los estudiantes, destacando que el 35.3% que percibe Python como más fácil también lo prefiere como método, mientras que el 29.4% que lo considera más difícil opta por métodos tradicionales.

Esta visualización revela una asociación: el 35.3% de estudiantes que perciben Python como más fácil también lo prefieren como método de resolución, mientras que el 29.4% que lo considera más difícil opta por el método tradicional de papel y lápiz. Destaca especialmente la ausencia total (0%) de estudiantes que, percibiendo Python como difícil, lo prefieran sobre el método tradicional. Esta marcada correlación evidencia cómo la percepción de facilidad constituye un factor determinante en la aceptación y adopción de Python como herramienta para el aprendizaje matemático, sugiriendo que las intervenciones educativas orientadas a reducir la percepción de dificultad podrían impactar positivamente en la disposición de los estudiantes hacia esta herramienta tecnológica.

4.7.3.2. Relación entre aprendizajes reportados y percepción de interés

Los estudiantes que reportaron aprendizajes significativos (como el manejo de variables y la resolución de problemas complejos) tendieron a mostrar mayor interés en aspectos como la automatización y las nuevas formas de resolución. Por ejemplo, el Estudiante 4, que mencionó haber aprendido sobre "las variables", también destacó como interesante "como los resultados se resolvían solos", estableciendo una conexión cognitiva entre el concepto aprendido y su aplicación práctica.

4.7.4. Implicaciones Pedagógicas y Conexión con la Teoría

4.7.4.1. Desarrollo del pensamiento matemático a través de Python

Los datos analizados muestran evidencia de cómo Python contribuye al desarrollo de diferentes tipos de pensamiento matemático según la clasificación del MEN (1998):

Pensamiento numérico: Los estudiantes reportan mayor facilidad para realizar operaciones complejas: "Lo que aprendimos fue a multiplicar con números demasiados grandes" (E14).

Pensamiento variacional: La comprensión de variables emerge como un aprendizaje significativo: "Las variables" (E4).

Pensamiento métrico: Aplicaciones a contextos de medición: "Fue calcular el costo de alimentos" (E3).

Pensamiento aleatorio: Organización y análisis de datos: "Planificar datos" (E3).

Esta distribución de impactos en diferentes tipos de pensamiento matemático confirma lo planteado por Weintrop et al. (2016) sobre la relación entre pensamiento computacional y pensamiento matemático, donde ambos se complementan y refuerzan mutuamente.

4.7.4.2. Barreras y facilitadores en la implementación de Python

El análisis permite identificar factores que actúan como barreras o facilitadores en la implementación de Python como herramienta de aprendizaje matemático:

Barreras identificadas:

Limitaciones técnicas: "los computadores no funcionan muy bien y dificultan el trabajo" (E1)

Curva de aprendizaje inicial: "Muy difícil porque no es fácil aprender a usar las variables" (E1)

Resistencia al cambio metodológico: "Prefiero por ahora papel y lápiz porque no me desenvuelvo muy bien en la página" (E1)

Facilitadores identificados:

Automatización de procesos: "me resulto un poquito más fácil porque solo poníamos los problemas y al lado aparecía los procesos solo para calcular" (E9)

Trabajo colaborativo: "me resulto más fácil por trabajar en grupo" (E11)

Retroalimentación inmediata: "Que fue divertido y más entendible ya que al lado salen los resultados" (E9)

Estos factores coinciden con los desafíos y oportunidades señalados por Hegedus & Moreno-Armella (2009) respecto a la implementación de tecnología en la educación matemática, confirmando que la tecnología debe transformar el pensamiento matemático y no simplemente replicar viejos estilos pedagógicos.

CAPÍTULO 5

5. CONCLUSIONES

5.1. Cumplimiento de los objetivos de investigación

La presente investigación logró caracterizar el Pensamiento Matemático que manifiestan los estudiantes de básica secundaria de la Escuela Normal Superior de Quibdó cuando resuelven tareas matemáticas usando Python. El enfoque fenomenológico adoptado permitió comprender en profundidad las experiencias de los estudiantes, mostrando cómo actúan al afrontar los problemas y cómo experimentan el proceso de trabajar con matemáticas apoyándose en la programación.

Con respecto al primer objetivo específico, se diseñaron tareas matemáticas relacionadas con la vida diaria de Quibdó que funcionaron muy bien para conocer las características del pensamiento matemático de los estudiantes. Las tareas trabajaron los cinco tipos de pensamiento matemático que establece el MEN (1998) y se adaptaron a cada nivel educativo.

El segundo objetivo se cumplió al analizar cómo los estudiantes usan y entienden los conceptos matemáticos cuando programan en Python. Se encontró que hay una evolución clara: los estudiantes de sexto grado trabajan de manera más concreta y específica, mientras que los de décimo grado pueden trabajar con ideas más abstractas y complejas. Los estudiantes mostraron diferentes niveles según el tipo de pensamiento matemático: mejor desarrollo en pensamiento numérico y variacional, desarrollo intermedio en pensamiento espacial y métrico, y menor desarrollo en pensamiento aleatorio.

El tercer objetivo se logró al identificar y describir los problemas de significado que tienen los estudiantes cuando trabajan con Python. Estos problemas no son algo malo, sino momentos importantes donde se construye el conocimiento y donde se encuentran las matemáticas tradicionales con la programación. Los problemas más importantes tienen que ver con las

diferencias entre escribir matemáticas de forma tradicional y escribir en Python, y también con la diferencia entre entender un concepto y poder programarlo.

5.2. Características principales del pensamiento matemático encontradas

La investigación mostró que el pensamiento matemático de los estudiantes cuando usan Python se puede describir a través de cuatro aspectos que cambian de manera diferente según el grado:

Cómo representan las ideas matemáticas: Los estudiantes van desarrollando poco a poco la capacidad para usar diferentes formas de representar las matemáticas. En sexto grado usan representaciones directas y específicas, mientras que en décimo grado pueden combinar mejor diferentes formas de mostrar las matemáticas, incluyendo estructuras de programación más complejas. Python funciona como una nueva forma de escribir matemáticas que enriquece, pero también hace más difícil la forma de representar ideas matemáticas.

Cómo resuelven problemas paso a paso: Se ve una evolución desde procedimientos simples y lineales hacia formas más organizadas de hacer algoritmos. Los estudiantes de sexto grado muestran pasos muy separados entre sí, mientras que los de décimo grado pueden hacer cadenas de pasos más conectadas y coherentes. Sin embargo, verificar y comprobar si los procedimientos están bien sigue siendo algo poco desarrollado en todos los niveles.

Cómo entienden los conceptos matemáticos: Los conceptos matemáticos atraviesan un proceso progresivo en el que los estudiantes transforman procedimientos en objetos que pueden manipular y relacionar. A medida que avanzan en su escolaridad, la organización del conocimiento se vuelve más clara, con mayor capacidad para estructurar conceptos complejos y establecer vínculos entre sus componentes.

Cómo reflexionan sobre su propio pensamiento: Este aspecto muestra el menor nivel de desarrollo en todos los grados, aunque se observa una evolución gradual. Los estudiantes empiezan a adquirir la capacidad de evaluar si sus soluciones tienen coherencia y de analizar sus propios procesos de pensamiento, pero estas habilidades aún requieren un fortalecimiento pedagógico más sistemático.

5.3. Cambios importantes según el nivel educativo

La investigación encontró cambios importantes en la forma básica del pensamiento matemático según avanza el nivel educativo. En sexto grado domina un pensamiento matemático concreto y específico, que se caracteriza por trabajar directamente con valores específicos y usar ejemplos muy relacionados con su experiencia personal. En octavo grado aparece un pensamiento más formal pero que aún está en desarrollo, con mayor capacidad para estructurar conceptos matemáticos, aunque con dificultades para implementarlos completamente en programación. En décimo grado se muestra un pensamiento más integrado y sistemático, con capacidad para estructurar datos complicados y crear modelos de situaciones complejas.

Estos cambios implican variaciones en cómo los estudiantes piensan, representan y trabajan con objetos matemáticos, más que simples cambios de cantidad. Python actúa como un impulsor de estos cambios al dar un medio para hacer visible y manipulable el pensamiento matemático.

Además, se encontró que los diferentes tipos de pensamiento matemático no evolucionan de la misma manera. El pensamiento numérico mostró mayor desarrollo en todos los grados, seguido por el pensamiento variacional. El pensamiento espacial y métrico tuvieron desarrollo intermedio, mientras que el pensamiento aleatorio fue el que menos se desarrolló. Esta diferencia refleja tanto la estructura del currículo tradicional como las oportunidades específicas que ofrece Python para trabajar ciertos tipos de pensamiento matemático.

5.4. Python como herramienta que cambia la forma de pensar matemáticas

Los resultados confirman que Python no es solo una calculadora, sino una herramienta que cambia de manera importante los procesos de pensamiento matemático. Como señalan Hegedus y Moreno-Armella (2009), la tecnología está aquí para transformar el pensamiento, y esta investigación da evidencia específica de cómo pasa esta transformación en el contexto educativo de Quibdó.

Python ayuda a hacer visible el pensamiento matemático, permitiendo que algoritmos y conceptos se conviertan en objetos que se pueden analizar y cambiar directamente. Desarrolla formas de representación más complejas al exigir que los estudiantes piensen sobre cómo representar conceptos matemáticos usando símbolos de programación. Facilita exploraciones sistemáticas que serían muy difíciles de hacer a mano, especialmente en niveles más avanzados.

Sin embargo, también crea tensiones importantes, problemas con la sintaxis en etapas iniciales, conflictos entre la forma tradicional de escribir matemáticas y la programación, y posible sobrecarga mental cuando los estudiantes deben prestar atención al mismo tiempo a aspectos matemáticos y a requisitos de programación.

Los estudiantes de sexto grado experimentaron principalmente Python como una herramienta para automatizar cálculos, con expresiones como "Como los resultados se resolvían solos" y "Que nos dio el resultado exacto sin error y sin dificultades". Esta percepción refleja una comprensión inicial de Python como calculadora avanzada, pero también muestra el potencial para liberar recursos cognitivos que pueden dirigirse hacia aspectos conceptuales más profundos.

En octavo grado, los estudiantes comenzaron a reconocer Python como un medio para expresar estructuras matemáticas más complejas, evidenciado en el uso de variables con nombres

descriptivos como "RAIZ_CUADRADA_DE_SUMA__CUADRADOS". Aunque estas implementaciones frecuentemente quedaron incompletas, demuestran una evolución hacia la comprensión de Python como lenguaje de expresión matemática.

Los estudiantes de décimo grado mostraron la integración más avanzada, usando Python para manejar estructuras de datos complejas e implementar transformaciones sistemáticas como la conversión de temperaturas. Sus códigos reflejan comprensión de Python como herramienta para modelar y analizar fenómenos complejos del mundo real.

5.5. Experiencias y sentimientos de los estudiantes

El enfoque fenomenológico adoptado permitió conocer las experiencias personales de los estudiantes, mostrando una gran variedad en percepciones y vivencias. Las entrevistas finales mostraron que el 47.1% de los estudiantes experimentó Python como algo que facilita resolver problemas matemáticos, mientras que el 35.3% lo sintió más difícil que los métodos tradicionales. Esta división refleja diferencias individuales importantes en cómo los estudiantes viven la integración entre pensamiento matemático y programación.

Los aspectos que más valoraron los estudiantes fueron la facilidad y automatización para resolver operaciones (29.4%) y el descubrimiento de nuevas formas de resolver problemas (23.5%). Estas percepciones confirman el potencial de Python para diversificar las estrategias de resolución de problemas y para desarrollar lo que Kilpatrick et al. (2001) llaman "competencia estratégica".

Los aprendizajes más importantes que reportaron se relacionan con entender variables y conceptos de programación (23.5%), aspecto que coincide con los hallazgos de Supelano Mesa (2023) sobre cómo Python facilita el paso de procesos aritméticos a algebraicos. Sin embargo, es notable que el mismo porcentaje de estudiantes (23.5%) dijo no haber aprendido nuevas

matemáticas, lo que sugiere la necesidad de estrategias educativas más claras para conectar la programación con el aprendizaje matemático.

Las dificultades más reportadas estuvieron relacionadas con el manejo de variables (29.4%), seguidas por las primeras interacciones con la plataforma (17.6%). La estrategia más usada para resolver estas confusiones fue pedir ayuda al profesor, lo que refuerza la importancia del rol del docente como mediador en la introducción de herramientas tecnológicas.

En cuanto a las preferencias metodológicas, se encontró una división: 41.2% prefiere Python y 41.2% prefiere papel y lápiz, con 17.6% sin preferencia clara. Esta división sugiere que la aceptación de Python como herramienta matemática está fuertemente relacionada con la percepción inicial de dificultad o facilidad.

5.6. Dificultades y obstáculos encontrados

La investigación identificó dificultades permanentes que necesitan atención educativa específica. La diferencia entre escribir matemáticas de forma tradicional y programar se muestra en todos los niveles, desde errores simples como confundir metros con metros cuadrados en sexto grado, hasta confusiones más complicadas como usar el operador módulo en lugar de división en décimo grado.

La diferencia entre entender conceptos y poder implementarlos en programación persiste, aunque se reduce poco a poco. Los estudiantes frecuentemente entienden conceptos matemáticos, pero tienen dificultades para implementarlos de manera funcional en Python. Este fenómeno, descrito por Schoenfeld (1992) como diferencia entre conocimiento conceptual y procedimental, necesita estrategias educativas que aborden directamente esta brecha.

La poca verificación y validación de resultados representa una oportunidad para desarrollar capacidades de reflexión sobre el propio pensamiento. Los estudiantes rara vez implementan

mecanismos claros para verificar si sus resultados son correctos, aspecto crucial para desarrollar pensamiento matemático crítico.

Otras dificultades incluyeron problemas técnicos con los equipos de cómputo, especialmente reportados por estudiantes que mencionaron que "los computadores eliminan el trabajo que ya hiciste y te toca empezar de nuevo". Estas limitaciones tecnológicas afectaron la experiencia de aprendizaje y sugieren la necesidad de mejorar la infraestructura tecnológica para optimizar el uso educativo de Python.

También se identificaron dificultades relacionadas con el idioma, como mencionó un estudiante de décimo grado: "lo más difícil fue lo de las letras en inglés pues la programación me pareció muy bien". Esto sugiere la necesidad de estrategias específicas para ayudar a los estudiantes a familiarizarse con la terminología en inglés usada en programación.

5.7. Diferencias según el tipo de pensamiento matemático

La caracterización reveló que los cinco tipos de pensamiento matemático establecidos por el MEN (1998) se manifiestan de manera diferente cuando los estudiantes usan Python:

Pensamiento numérico: Fue el más desarrollado en todos los niveles. Los estudiantes mostraron facilidad para trabajar con operaciones básicas, conversiones de unidades, y cálculos con números grandes. En sexto grado se enfocaron en operaciones directas, en octavo grado reconocieron patrones numéricos importantes como la terna pitagórica 3-4-5, y en décimo grado integraron el pensamiento numérico en contextos aplicados complejos como análisis económico.

Pensamiento variacional: Mostró el segundo mejor desarrollo. En sexto grado se manifestó de forma implícita con comprensión básica de proporcionalidad. En octavo grado apareció mayor reconocimiento de relaciones funcionales, aunque con implementación limitada.

En décimo grado se desarrolló altamente, con capacidad para trabajar con series de tiempo, transformaciones funcionales y relaciones sistemáticas entre variables.

Pensamiento espacial: Tuvo desarrollo intermedio. En sexto grado mostró comprensión operativa de conceptos básicos como área y perímetro, pero con limitada comprensión dimensional. En octavo grado apareció formalización conceptual del Teorema de Pitágoras con comprensión de su estructura, aunque con implementación incompleta. En décimo grado no se evidenció directamente en los códigos analizados.

Pensamiento métrico: También con desarrollo intermedio. En sexto grado mostró comprensión básica de unidades y conversiones simples. En octavo grado apareció mayor capacidad para trabajar con medidas derivadas y relaciones entre diferentes tipos de magnitudes. En décimo grado mostró integración sofisticada de diferentes sistemas de medición en contextos de análisis de datos reales.

Pensamiento aleatorio: Fue el menos desarrollado en todos los niveles. En sexto grado estuvo prácticamente ausente en los códigos analizados. En octavo grado apareció de forma emergente en algunos contextos, aunque sin implementación algorítmica sustancial. En décimo grado mostró mayor desarrollo, evidenciado en la capacidad para organizar y analizar datos sistemáticamente, aunque con implementación estadística aún limitada.

5.8. Implicaciones para la educación matemática

Los hallazgos tienen implicaciones directas para la práctica educativa en matemáticas. Primero, la necesidad de crear puentes claros entre la forma tradicional de escribir matemáticas y la sintaxis de programación. Esto implica diseñar actividades que establezcan conexiones sistemáticas entre diferentes formas de representación.

Segundo, la importancia de introducir los elementos de Python de manera gradual y estratégica, empezando con estructuras más parecidas a la notación matemática tradicional y avanzando poco a poco hacia construcciones más específicas de programación.

Tercero, la necesidad de estrategias diferentes según el nivel educativo: promover generalización progresiva en sexto grado, integrar conceptos con procedimientos en octavo grado, y refinar aspectos sintácticos específicos en décimo grado.

Cuarto, el desarrollo intencional de capacidades para reflexionar sobre el propio pensamiento a través de prácticas sistemáticas de documentación reflexiva, actividades de análisis de errores, y comparación de diferentes soluciones.

Las implicaciones también sugieren la importancia de abordar las limitaciones tecnológicas identificadas. La falta de computadores funcionales y conectividad estable afectó la experiencia de aprendizaje. Como mencionó un estudiante: "Mas difícil ya que los computadores no funcionan muy bien y dificultan el trabajo". Esto indica que la implementación exitosa de Python en educación matemática requiere estrategias pedagógicas apropiadas, e infraestructura tecnológica adecuada.

5.9. Aportes al conocimiento sobre educación matemática

Esta investigación aporta al conocimiento teórico sobre la integración de programación en la educación matemática al proporcionar un marco de cuatro dimensiones para caracterizar el pensamiento matemático mediado por tecnología. Las cuatro dimensiones identificadas (representacional-simbólica, procedimental-estratégica, conceptual-estructural, y metacognitiva-reflexiva) ofrecen un esquema de análisis que puede aplicarse en otros contextos y con otras herramientas tecnológicas.

Metodológicamente, la investigación demuestra que el enfoque fenomenológico es apropiado para estudios de integración tecnológica en educación matemática. Este enfoque permite capturar aspectos cualitativos de la experiencia de aprendizaje que frecuentemente se pierden en aproximaciones solo cuantitativas o centradas únicamente en resultados de desempeño.

La combinación sistemática de análisis de códigos Python, entrevistas en profundidad, y observaciones de clase proporcionó una comprensión rica y detallada de cómo los estudiantes experimentan y muestran pensamiento matemático mediante programación.

Los resultados también aportan evidencia empírica específica sobre cómo la tecnología transforma el pensamiento matemático en contextos educativos reales, complementando las propuestas teóricas de autores como Borba et al. (2016) sobre la relación entre medios y conocimiento.

5.10. Limitaciones del estudio

Esta investigación presenta limitaciones que abren oportunidades para investigaciones futuras. El tiempo limitado de implementación (seis semanas) no permitió observar desarrollos a más largo plazo en el pensamiento matemático de los estudiantes. Estudios más largos podrían revelar patrones de desarrollo más estables y sostenibles.

Las limitaciones tecnológicas enfrentadas en la Escuela Normal Superior de Quibdó (falta de computadores funcionales, conectividad intermitente) requirieron adaptaciones metodológicas que podrían haber afectado los resultados. Investigaciones en contextos con mejores recursos tecnológicos podrían revelar potencialidades adicionales de la integración Python-matemáticas.

El estudio se centró en tareas específicamente diseñadas para la investigación. Estudios futuros podrían explorar cómo se manifiesta el pensamiento matemático cuando los estudiantes utilizan Python para abordar tareas curriculares regulares o problemas auténticos de su entorno.

La muestra se limitó a tres grados específicos (sexto, octavo y décimo) de una sola institución. Investigaciones con muestras más amplias que incluyan otros grados y otras instituciones podrían proporcionar una caracterización más completa del pensamiento matemático mediado por Python.

Una limitación metodológica importante de este estudio se relaciona con la inferencia de procesos cognitivos a partir del análisis de códigos Python. Aunque las interpretaciones sobre pensamiento matemático están fundamentadas en marcos teóricos establecidos y se apoyan en triangulación con otras fuentes de datos, las inferencias sobre procesos mentales internos representan construcciones analíticas del investigador más que confirmaciones directas de los procesos cognitivos de los estudiantes.

Futuras investigaciones podrían beneficiarse de la implementación de protocolos de pensamiento en voz alta durante la programación, entrevistas cognitivas específicas sobre decisiones de codificación, o estudios longitudinales que permitan validar las interpretaciones sobre desarrollo del pensamiento matemático a través de múltiples momentos y contextos.

Esta limitación no invalida los hallazgos presentados, pero sugiere la importancia de interpretar las caracterizaciones del pensamiento matemático como marcos analíticos fundamentados que contribuyen a la comprensión del fenómeno estudiado, reconociendo que la confirmación definitiva de procesos cognitivos específicos requeriría aproximaciones metodológicas adicionales.

5.11. Recomendaciones para futuras investigaciones

Basándose en los hallazgos y limitaciones identificadas, se sugieren las siguientes líneas de investigación futura:

Estudios longitudinales que sigan a los mismos estudiantes durante períodos más largos para observar la evolución sostenida del pensamiento matemático mediado por Python. Estos estudios podrían revelar patrones de desarrollo que no son visibles en intervenciones cortas.

Investigaciones comparativas entre diferentes herramientas de programación (Python, Scratch, JavaScript) para determinar cuáles son más efectivas para desarrollar diferentes tipos de pensamiento matemático en diferentes niveles educativos.

Estudios sobre la formación docente necesaria para implementar efectivamente Python en la educación matemática, considerando tanto aspectos técnicos como pedagógicos.

Investigaciones sobre el diseño de currículos integrados que combinen sistemáticamente objetivos de aprendizaje matemático con competencias de programación, superando la fragmentación actual entre estas áreas.

Estudios sobre la transferencia de aprendizajes: cómo las habilidades desarrolladas con Python se transfieren a la resolución de problemas matemáticos sin tecnología y viceversa.

Investigaciones sobre la implementación de Python en diferentes contextos socioeconómicos para entender cómo factores como acceso tecnológico, apoyo familiar, y recursos institucionales afectan los resultados.

5.12. Reflexiones finales

Esta investigación confirma que caracterizar el pensamiento matemático mediante Python revela aspectos importantes sobre cómo los estudiantes piensan y aprenden matemáticas. En el contexto específico de la Escuela Normal Superior de Quibdó, Python emerge como una herramienta con potencial significativo para enriquecer la experiencia de aprendizaje matemático.

Sin embargo, este potencial no se realiza automáticamente por introducir la herramienta tecnológica. Requiere una comprensión profunda de cómo los estudiantes experimentan la integración entre pensamiento matemático y programación, así como estrategias educativas específicas que aborden las dificultades identificadas.

La caracterización multidimensional del pensamiento matemático desarrollada en esta investigación proporciona una base sólida para entender mejor cómo los estudiantes aprenden matemáticas con tecnología. Como señala Borba et al. (2016), si los medios cambian, el conocimiento también puede cambiar, y esta investigación documenta cómo ocurre este cambio en el contexto particular de estudiantes chocoanos aprendiendo matemáticas con Python.

Los hallazgos sugieren que la caracterización del pensamiento matemático mediado por Python revela cómo los estudiantes aprenden conceptos matemáticos específicos, y cómo desarrollan formas de pensamiento más flexibles, sistemáticas y reflexivas que serán valiosas para su desarrollo académico y profesional futuro. En un mundo cada vez más digitalizado, estas competencias integradas matemático-computacionales se vuelven esenciales para la participación ciudadana plena y el desarrollo económico regional.

La experiencia fenomenológica de los estudiantes, capturada a través de esta investigación, muestra que más allá de los aspectos técnicos y conceptuales, el uso de Python en matemáticas genera emociones, motivaciones y perspectivas que influyen en cómo los estudiantes se relacionan con las matemáticas. Esta dimensión experiencial del aprendizaje matemático, frecuentemente ignorada en aproximaciones más tradicionales, emerge como un factor crucial para el éxito de cualquier innovación educativa en el área.

Anexos

Anexo 1. Guía de Reconocimiento de Objetos y Significados (GROS) Matemáticos para las tareas propuestas.

Tipos de Objetos	Significados (relación de referencia o de uso)
REPRESENTACIONES (Términos y expresiones matemáticas; símbolos, representaciones gráficas)	
Encontrar ceros de una función polinómica	Encontrar los valores que, al reemplazar la variable independiente, anulan la función. Por ejemplo, en décimo grado: análisis de costos vs ingresos en el mercado.
Encontrar período de una función trigonométrica	Determinar el intervalo mínimo en que la función repite sus valores. Aplicado en décimo: análisis de ciclos de cultivo.
Identificar amplitud de una función trigonométrica	Determinar la altura de la oscilación de la función. Contextualizado en variaciones de temperatura diaria.
Identificar desplazamientos de una función trigonométrica	Determinar traslaciones horizontales o verticales de la función. Aplicado al análisis de patrones climáticos.
CONCEPTOS (Entidades matemáticas para las cuales se puede formular una definición)	
MCM y MCD	En sexto grado: aplicado a horarios de lanchas y planificación de viajes.
Función Polinómica	Expresión que modela costos y ganancias en el mercado local.
Función Trigonométrica	Modelado de ciclos naturales y comerciales.
Variable independiente	En contextos de tiempo, cantidad de productos, número de clientes.
Variable dependiente	Precios, ganancias, producción según variables independientes.
PROCEDIMIENTOS (Técnicas, operaciones, algoritmos)	
Representación gráfica de una función	Visualización de datos de ventas, crecimiento de cultivos, patrones climáticos.
Encontrar raíces de una función polinómica	Determinar puntos de equilibrio en costos y ganancias.
Encontrar período de una función trigonométrica	Análisis de ciclos comerciales y naturales.
PROPIEDADES (Enunciados que requieren justificación o prueba)	
Teorema del álgebra	Aplicado al análisis de funciones de costo y ganancia.
Relaciones trigonométricas	Uso en diseño de construcciones y mediciones indirectas.
Paridad de las funciones	Análisis de simetrías en patrones comerciales y naturales.

Anexo 2. Guía de Reconocimiento de Objetos y Significados (GROS)
Computacionales para las tareas propuestas usando Python.

Tipos de Objetos	Significados (relación de referencia o de uso)
REPRESENTACIONES	
Comando de Python	Instrucciones para cálculos comerciales, análisis de datos y modelado matemático.
Representación gráfica en Python	Visualización de datos de ventas, crecimiento y patrones climáticos.
CONCEPTOS	
Librería NumPy	Para cálculos con datos del mercado y análisis estadístico.
Matplotlib	Visualización de datos comerciales y ambientales.
Math	Operaciones matemáticas básicas en contextos comerciales.
Sympy	Cálculos simbólicos para modelado de situaciones complejas.
PROCEDIMIENTOS	
Graficar una función usando Python	Representación visual de datos comerciales y ambientales.
Factorizar un polinomio con Python	Análisis de funciones de costo y ganancia.
Encontrar raíces con Python	Determinación de puntos de equilibrio económico.

Anexo 3. Tabla de características y objetos matemáticos a examinar en las tareas propuestas.

Características a observar en las soluciones de los estudiantes usando Python	Presente
Python	
Uso de librerías (NumPy, Matplotlib, Math, Sympy)	X
Definición de variables	X
Uso de funciones y comandos para:	
- Graficar	X
- Factorizar	X
- Encontrar raíces	X
- Hacer zoom	X
Función Polinómica	
Identificación del grado	X
Representación algebraica	X
Identificación de raíces	X
Representación gráfica	X
Función Trigonométrica	
Identificación del período	X
Identificación de la amplitud	X
Identificación de desplazamientos	X
Representación algebraica	X
Representación gráfica	X

Anexo 4. Guías de Reconocimiento de Objetos y Significados (GROS) Matemáticos para las tareas propuestas en cada tipo de pensamiento matemático.

Tabla 1. GROS para el pensamiento numérico y sistemas numéricos

Tipos de Objetos	Significados (relación de referencia o de uso)
REPRESENTACIONES	
Números primos (6° grado)	Números naturales para análisis de divisibilidad en contextos comerciales
MCM y MCD (6° grado)	Planificación de horarios y distribución de productos
Notación científica (8° grado)	Manejo de cantidades muy grandes o pequeñas en mediciones
Porcentajes (10° grado)	Análisis de ganancias, pérdidas e intereses
CONCEPTOS	
Divisibilidad (6° grado)	Distribución equitativa de productos
Proporcionalidad (8° grado)	Relaciones entre cantidades en el comercio
Tasas de interés (10° grado)	Análisis de préstamos y costos financieros
PROCEDIMIENTOS	
Cálculo de MCM (6° grado)	Determinación de ciclos comerciales
Conversiones (8° grado)	Transformación entre diferentes unidades
Cálculo financiero (10° grado)	Análisis de opciones de préstamo

Tabla 2. GROS para el pensamiento espacial y sistemas geométricos

Tipos de Objetos	Significados (relación de referencia o de uso)
REPRESENTACIONES	
Figuras planas (6° grado)	Diseño de espacios comerciales
Teorema de Pitágoras (8° grado)	Construcción de estructuras
Optimización espacial (10° grado)	Diseño eficiente de almacenamiento
CONCEPTOS	
Área y perímetro (6° grado)	Medición de espacios comerciales
Ángulos rectos (8° grado)	Construcción de estructuras
Volumen óptimo (10° grado)	Diseño de contenedores y espacios
PROCEDIMIENTOS	
Medición directa (6° grado)	Cálculo de espacios comerciales
Construcciones geométricas (8° grado)	Diseño de estructuras
Optimización (10° grado)	Maximización de espacio útil

Tabla 3. GROS para el pensamiento métrico y sistemas de medidas

Tipos de Objetos	Significados (relación de referencia o de uso)
REPRESENTACIONES	
Unidades básicas (6° grado)	Medidas de peso y longitud en el comercio
Temperatura y humedad (8° grado)	Control ambiental en cultivos
Volumen y capacidad (10° grado)	Diseño de contenedores y tanques
CONCEPTOS	
Conversiones simples (6° grado)	Entre unidades comerciales comunes
Medidas ambientales (8° grado)	Monitoreo de condiciones de cultivo
Densidad y capacidad (10° grado)	Optimización de almacenamiento
PROCEDIMIENTOS	
Medición de peso (6° grado)	Control de productos comerciales
Monitoreo ambiental (8° grado)	Seguimiento de condiciones de cultivo
Diseño de contenedores (10° grado)	Optimización de almacenamiento

Tabla 4. GROS para el pensamiento aleatorio y sistemas de datos

Tipos de Objetos	Significados (relación de referencia o de uso)
REPRESENTACIONES	
Gráficos simples (6° grado)	Visualización de ventas diarias
Probabilidad (8° grado)	Análisis de patrones climáticos
Regresión lineal (10° grado)	Predicción de ventas y tendencias
CONCEPTOS	
Promedio (6° grado)	Análisis básico de ventas
Probabilidad condicional (8° grado)	Predicción de eventos climáticos
Correlación (10° grado)	Relaciones entre variables comerciales
PROCEDIMIENTOS	
Cálculos básicos (6° grado)	Análisis simple de datos comerciales
Estimación de probabilidad (8° grado)	Predicción de condiciones
Análisis estadístico (10° grado)	Modelado de tendencias comerciales

Tabla 5. GROS para el pensamiento variacional y sistemas algebraicos y analíticos

Tipos de Objetos	Significados (relación de referencia o de uso)
REPRESENTACIONES	
Proporciones (6° grado)	Relaciones precio-cantidad
Funciones lineales (8° grado)	Modelado de costos simples
Funciones de crecimiento (10° grado)	Modelado de cultivos y ventas
CONCEPTOS	
Proporcionalidad (6° grado)	Relaciones comerciales básicas
Ecuaciones lineales (8° grado)	Modelado de situaciones comerciales
Modelos de crecimiento (10° grado)	Análisis de desarrollo de cultivos
PROCEDIMIENTOS	
Regla de tres (6° grado)	Cálculos comerciales simples
Resolución de ecuaciones (8° grado)	Análisis de situaciones comerciales
Modelado matemático (10° grado)	Análisis de sistemas complejos

Anexo 5. Solicitud de autorización al rector de la Escuela Normal Superior de Quibdó para la implementación del estudio.

Quibdó, 26 de septiembre de 2024

Señor:

Emiro Borja Palacios

Rector Escuela Normal Superior de Quibdó

Quibdó - Chocó

Asunto: Autorización para la implementación del estudio de investigación

Cordial saludo.

Por medio de la presente, yo, Kevin Estiguar Palacios Maturana, identificado con cédula 1.003.970.071 estudiante de la Maestría en Educación de la Universidad de Antioquia, me dirijo a usted de manera respetuosa para solicitar su autorización para implementar mi estudio de investigación titulado “Caracterización del Pensamiento Matemático de estudiantes de secundaria cuando resuelven tareas matemáticas mediante Python”, con el fin de recolectar los datos necesarios para el desarrollo de mi tesis de maestría.

El estudio está programado para ser realizado con una muestra de estudiantes de básica secundaria de la Escuela Normal Superior de Quibdó y será llevado a cabo durante el segundo semestre del año 2024 en las instalaciones de la institución educativa, específicamente en el laboratorio de computación o sala de informática. De igual manera, solicito su autorización para contar con la participación de los estudiantes seleccionados en las actividades de recolección de datos, las cuales incluyen entrevistas, grupos focales, observaciones y análisis de trabajos realizados con Python, sin que esto afecte sus compromisos académicos.

Adjunto a esta solicitud encontrará el aval del Asesor de la Investigación. La Investigación se realiza en cumplimiento de los Estándares Éticos de la Universidad de Antioquia.

Es importante mencionar que se mantendrá la confidencialidad de los datos personales de todos los participantes y que los resultados del estudio serán utilizados únicamente con fines académicos. Además, me comprometo a socializar los hallazgos de la investigación con la comunidad educativa de la Escuela Normal Superior de Quibdó una vez finalizado el estudio.

Agradezco de antemano su valiosa colaboración y quedo atento a su respuesta.

Atentamente,

Kevin Estiguar Palacios Maturana

Autoriza:

Emiro Borja Palacios

Rector Escuela Normal Superior de Quibdó

Anexo 6. Formato de consentimiento informado para padres de familia y/o acudientes de los estudiantes participantes.

INFORMACIÓN PREVIA A LA INVESTIGACIÓN

Investigador: Kevin Estiguar Palacios Maturana

Institución Educativa: Escuela Normal Superior de Quibdó

Estimado representante legal,

A través del presente documento, nos gustaría contar con su autorización para que el estudiante a quien representa participe en la investigación titulada: “*Caracterización del Pensamiento Matemático de estudiantes de secundaria cuando resuelven tareas matemáticas mediante Python*”. Antes de tomar la decisión, es necesario que comprenda en qué consiste este estudio. Por favor, lea cuidadosamente la siguiente información y realice las preguntas correspondientes sobre cualquier aspecto que no comprenda.

1. ¿Dónde se llevará a cabo esta investigación?

La investigación se realizará en la Escuela Normal Superior de Quibdó, en las instalaciones y laboratorios de computación de la institución.

2. ¿Cuál es el objetivo de esta investigación?

Este estudio forma parte del trabajo de grado para optar al título de Magíster en Educación de la Universidad de Antioquia. El objetivo es caracterizar el Pensamiento Matemático manifestado por estudiantes de básica secundaria cuando resuelven tareas matemáticas mediante el lenguaje de programación Python.

3. ¿Por qué es importante esta investigación?

Esta investigación es importante porque permitirá comprender cómo el uso de Python puede promover el desarrollo del Pensamiento Matemático en estudiantes de secundaria, identificando fortalezas, debilidades y oportunidades en el proceso de enseñanza-aprendizaje de las matemáticas.

4. ¿Por qué ha sido invitado el estudiante a participar?

El estudiante ha sido invitado a participar porque cumple con los criterios de inclusión: ser estudiante de básica secundaria de la Escuela Normal Superior de Quibdó y estar cursando grados donde se imparten contenidos matemáticos.

5. ¿En qué consistirá la participación y cuánto durará?

La participación consistirá en resolver tareas matemáticas utilizando Python durante las clases regulares. Además, se realizarán entrevistas, grupos focales y se analizarán los trabajos realizados por los estudiantes. La duración de la investigación será durante el segundo semestre del año 2024.

6. ¿Es posible retirarse de la investigación?

La participación es completamente voluntaria. Tanto los estudiantes como sus representantes legales tienen derecho a no participar o a retirarse en cualquier momento sin ninguna consecuencia.

7. ¿Cómo se recolectará y utilizará la información?

Los datos se recolectarán mediante entrevistas, registros audiovisuales, trabajos de los estudiantes, etc. Serán utilizados únicamente con fines académicos, garantizando la confidencialidad y el anonimato de los participantes.

Se cuenta con la aprobación de las directivas de la institución. Puede solicitar más información en cualquier momento. Al firmar, acepta la participación voluntaria del estudiante a quien representa bajo las condiciones expuestas.

CONSENTIMIENTO INFORMADO

Yo, _____ con C.C. _____ en calidad de representante legal del(la) estudiante _____ del grado _____, declaro que:

- He sido informado sobre los objetivos y procedimientos de esta investigación.
- Se me ha brindado la oportunidad de realizar preguntas y aclarar dudas.
- Tengo conocimiento que la participación es voluntaria y sin retribución económica.
- Puedo solicitar que el estudiante a quien represento se retire en cualquier momento sin consecuencias.
- Se garantizará la confidencialidad de los datos personales con fines estrictamente académicos.

Por lo anterior, acepto voluntariamente que el estudiante a quien represento participe en esta investigación.

Nombre del representante legal: _____

Firma del representante legal: _____

Nombre del estudiante: _____

Firma del estudiante: _____

Fecha: _____

Referencias bibliográficas

- Altamirano, D. A. L., Cunalata, O. G. T., Ortiz, R. G. B., Bayas, Á. F., Chimborazo, R. M. C., Chicaiza, W. O. G., ... & Jaya, P. E. M. (2023). Python una escalera para el desarrollo de la inteligencia artificial en el proceso enseñanza y aprendizaje de las matemáticas. *Dominio de las Ciencias*, 9(4), 363-374.
- Altamirano, E., García, M., & Moya, C. (2023). Python aplicado a la enseñanza de las matemáticas: Una revisión sistemática. *Revista de Educación Matemática*, 34(2), 367-382.
- Arotuma, C., & Soraya, S. (2017). La Programación como herramienta educativa-SCRATCH.
- Ayora Carchi, R. M. (2013). El razonamiento lógico matemático y su incidencia en el aprendizaje de los estudiantes de la escuela teniente hugo ortiz, de la comunidad zhizho, cantón cuenca, provincia del azuay [Tesis de pregrado]. Universidad de Cuenca.
- Borba, M., Askar, P., Engelbrecht, J., Gadaniadis, G., Llinares, S., & Sánchez-Aguilar, M. (2016). Blended learning, e-learning and mobile learning in mathematics education. *ZDM Mathematics Education*, 48(5), 589-610.
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77-101.
- Briz Redón, Á., & Serrano Aroca, Á. (2018). Aprendizaje de las matemáticas a través del lenguaje de programación R en Educación Secundaria. *Educación Matemática*, 30(1), 133-162.
- Brousseau, G. (1997). *Theory of didactical situations in mathematics*. Kluwer Academic Publishers.
- Bustamante Narváez, L. F., & Rodríguez Useche, V. M. (2022). Resolución de problemas a través de programación en Python centrado en la Teoría de Situaciones Didácticas dirigido a estudiantes de grado noveno.
- Cabero, J. (2001). *Tecnología educativa: Diseño y utilización de medios en la enseñanza*. Paidós.

- Castro . W.F.; Godino Juan D.; Rivas M (2011). Razonamiento algebraico en educación primaria: Un desafío para la formación de maestros. Actas del XII Encuentro Colombiano de Matemática Educativa. Armenia.
- Cevallos, M., & Cuadros, J. (2015). El material didáctico en matemática en el rendimiento lógico y cognitivo de básica elemental [Tesis de grado]. Universidad Laica Eloy Alfaro de Manabí.
- Challenger-Pérez, I., Díaz-Ricardo, Y., & Becerra-García, R. A. (2014). El lenguaje de programación Python. Ciencias Holguín, 20(2), 1-13.
- da Silva, M. (2024). Google Colaboratory como herramienta educativa: Perspectivas y aplicaciones. Journal of Educational Technology, 4(1), 1-12.
- Dainamang, S. A., Praherdhiono, H., & Soepriyanto, Y. (2024). Design of learning python programming for informatics education student using cloud computing technology based on google colaboratory. Journal of Research in Instructional, 4(1), 111-120.
- Denzin, N. K. (1978). The research act: A theoretical introduction to sociological methods. McGraw-Hill.
- Departamento Administrativo Nacional de Estadística. (2023). Pobreza multidimensional en Colombia 2022. DANE. <https://www.dane.gov.co/index.php/estadisticas-por-tema/pobreza-y-condiciones-de-vida/pobreza-multidimensional/pobreza-multidimensional-2022>
- Departamento Administrativo Nacional de Estadística. (2024). Pobreza monetaria en Colombia 2023. DANE. <https://www.dane.gov.co/index.php/estadisticas-por-tema/pobreza-y-condiciones-de-vida/pobreza-monetaria>
- Du Boulay, J. B. H. "Teaching teachers mathematics through programming." International Journal of Mathematical Educational in Science and Technology 11.3 (1980): 347-360.
- Duval, R. (1995). Sémiotique et pensée humaine: Registres sémiotiques et apprentissages intellectuels. Peter Lang.

- Flores Samaniego, Á. H. (2017). Pensamiento matemático y el quehacer científico. *Pädi: Revista de Proyectos y Textos Académicos en Didáctica de las Ciencias y la Ingeniería*, 1(1), 26-39.
- Freudenthal, H. (1991). *Revisiting mathematics education: China lectures*. Kluwer Academic Publishers.
- García-González, M., Torres-Rodríguez, C., & Martínez-Sierra, G. (2020). Tecnologías digitales en la educación matemática: Una revisión sistemática. *RELIME*, 23(1), 7-32.
- García-González, L. A., & Solano-Suarez, A. (2020). Enseñanza de la Matemática mediada por la tecnología. *EduSol*, 20(70), 84-99.
- Gardner, H. (1983). *Frames of mind: The theory of multiple intelligences*. Basic Books.
- Gholami, M., & Muehlemann, S. (2024). Math Skills, Selection in Training Firms, and Post-Training Wages (No. 0212). University of Zurich, Department of Business Administration.
- Gómez, A., & Inagan, J. (2014). Desarrollo del Razonamiento Cuantitativo a través de Scratch. XII Coloquio Regional de Matemáticas y II Simposio de Estadística.
- González, F. (2001). El pensamiento matemático de los niños: Un estudio constructivista. *Educación Matemática*, 13(2), 81-98.
- Goos, M., & Geiger, V. (2012). Connecting social perspectives on mathematics teacher education in online environments. *ZDM—The International Journal on Mathematics Education*, 44(6), 705-715.
- Hegedus, S. J., & Moreno-Armella, L. (2009). Introduction: The transformative nature of "dynamic" educational technology. *ZDM Mathematics Education*, 41, 397-398.
- Hernández Sampieri, R. (2014). *Metodología de la investigación* (6a ed.). McGraw-Hill.
- Hernández, R., Orrego, R., & Quiñones, S. (2018). Nuevas formas de aprender: La formación docente en el uso de las TIC. *Propósitos y Representaciones*, 6(2), 671-701.

- Hidalgo, M. I. M. (2018). Estrategias metodológicas para el desarrollo del pensamiento lógico-matemático. *Didasc@lia: Didáctica y Educación*, 9(1), 125-132.
- Horn, M. S., & Jacob, R. J. (2007). Tangible programming in the classroom with Tern. In CHI'07 extended abstracts on Human factors in computing systems (pp. 1965-1970). ACM. <https://doi.org/10.1145/1240866.1240933>
- Hoyles, C., Noss, R., Kent, P., & Bakker, A. (2010). Improving mathematics at work: The need for techno-mathematical literacies. Routledge.
- Inhelder, Bärbel, and Jean Piaget. The growth of logical thinking from childhood to adolescence: An essay on the construction of formal operational structures. Routledge, 2013.
- Instituto Colombiano para la Evaluación de la Educación (ICFES). (2022). Informe nacional de resultados Saber 11° – 2022. ICFES.
- Kaput, J., & Roschelle, J. (2013). The Mathematics of Change and Variation from a Millennial Perspective: New content, new context. Springer.
- Kilpatrick, J., Swafford, J., & Findell, B. (2001). Adding it up: Helping children learn mathematics. National Academy Press.
- Lehrer, R. (2003). Developing understanding of measurement. In J. Kilpatrick, W. G. Martin, & D. Schifter (Eds.), A research companion to principles and standards for school mathematics (pp. 179-192). National Council of Teachers of Mathematics.
- Lesh, R., & Doerr, H. M. (2003). Beyond constructivism: Models and modeling perspectives on mathematics problem solving, learning, and teaching. Lawrence Erlbaum Associates.
- Maggio, M. (2018). Habilidades del siglo XXI. Cuando el futuro es hoy. Fundación Santillana.
- Maldonado, M. (2010). Currículo con enfoque de competencias. ECOE Ediciones.
- Mason, J., Burton, L., & Stacey, K. (2010). Thinking mathematically (2nd ed.). Pearson.

- Mendoza Aguirre, M. (2018). Software de programación "Scratch" en el desarrollo del pensamiento lógico matemático de estudiantes de una institución educativa primaria, Chinchipe-2017 [Tesis de maestría]. Universidad César Vallejo.
- Ministerio de Educación Nacional. (1998). Lineamientos Curriculares de Matemáticas. MEN.
- Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. (2021, 18 de marzo). UNESCO: Las Matemáticas, enseñanza e investigación para enfrentar los desafíos de estos tiempos.
- Padilla, R. D. M. (2019). La llegada de la inteligencia artificial a la educación. Revista de Investigación en Tecnologías de la Información: RITI, 7(14), 260-270.
- Pertejo-López, J. (2017). Programación gráfica y robótica para fomentar la competencia matemática [Tesis de pregrado].
- Piaget, J., & Inhelder, B. (1969). The psychology of the child. Basic Books.
- Pierce, R., Stacey, K., Wander, R., & Ball, L. (2011). The design of lessons using mathematics analysis software to support multiple representations in secondary school mathematics. Technology, Pedagogy and Education, 20(1), 95-112.
- Ponte, J. P. (2005). Gestão curricular em Matemática. In GTI (Ed.), O professor e o desenvolvimento curricular (pp. 11-34). APM.
- Ponte, J. P., & Quaresma, M. (2012). O papel do contexto nas tarefas matemáticas. Interações, 8(22), 196-216.
- Reyes-Santander, P. (2014). Caracterización del Pensamiento Matemático: Escenarios con estudiantes universitarios y de liceo utilizando temas de la Teoría de Grupos.
- Rodríguez Vanegas, E. A., & Orjuela Díaz, M. (2022). Desarrollo del pensamiento variacional y computacional a través del lenguaje Python en Educación Básica Secundaria en Colombia [Monografía]. Universidad Nacional Abierta y a Distancia.

- Rouhiainen, L. (2018). *Inteligencia artificial: 101 cosas que debes saber hoy sobre nuestro futuro*. Alienta Editorial.
- Russell, S. J., Norvig, P., & Davis, E. (2010). *Artificial Intelligence* (3a ed.). Prentice Hall.
- Salgado, A. (2007). Investigación cualitativa: diseños, evaluación del rigor metodológico y retos. *Liberabit*, 13(13), 71-78.
- Sánchez, M. (1995). *Desarrollo de Habilidades del Pensamiento: Procesos Básicos del Pensamiento*. Trillas.
- Santos-Trigo, M., & Camacho-Machín, M. (2013). Framing the use of computational technology in problem solving approaches. *The Mathematics Enthusiast*, 10(1), 279-302.
- Schoenfeld, A. H. (1992). Learning to think mathematically: Problem solving, metacognition, and sense-making in mathematics. En D. Grouws (Ed.), *Handbook of research on mathematics teaching and learning* (pp. 334-370). Macmillan.
- Sokolowski, Andrzej, Yeping Li, and Victor Willson. "The effects of using exploratory computerized environments in grades 1 to 8 mathematics: A meta-analysis of research." *International Journal of STEM Education* 2.1 (2015): 8.
- Stein, M. K., & Smith, M. S. (2009). Mathematical tasks as a framework for reflection: From research to practice. *Mathematics Teaching in the Middle School*, 3(4), 268-275.
- Suárez, G. (2019). *Recursos educativos digitales en el desarrollo del pensamiento lógico matemático* [Tesis de pregrado]. Universidad de Guayaquil.
- Supelano Mesa, L. F. (2023). *Resolución de problemas geométricos y algebraicos a través de la programación usando el lenguaje Python en la Institución Educativa Departamental Rafael Pombo Sopó*.
- Tall, D., & Vinner, S. (1981). Concept image and concept definition in mathematics with particular reference to limits and continuity. *Educational Studies in Mathematics*, 12(2), 151-169.

- van Hiele, P. M. (1986). *Structure and insight: A theory of mathematics education*. Academic Press.
- Varela Uribe, A. (2021). Enseñanza individualizada de matemáticas mediante herramientas de inteligencia artificial en entornos escolares tradicionales.
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.
- Weintrop, D., & Wilensky, U. (2017). Comparing block-based and text-based programming in high school computer science classrooms. *ACM Transactions on Computing Education (TOCE)*, 18(1), 1-25. <https://doi.org/10.1145/3089799>
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. *Journal of Science Education and Technology*, 25, 127-147.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Yeo, J. B. W. (2017). Development of a framework to characterise the openness of mathematical tasks. *International Journal of Science and Mathematics Education*, 15, 175-191.
- Zerpa, Y. D. (2003). Habilidades de pensamiento matemático en alumnos de educación básica. *Revista de Pedagogía*, 24(71), 347-363.