

La importancia de las matemáticas en la estructura de datos: optimización y eficiencia

The importance of mathematics in data structure: optimization and efficiency

- 1 Lourdes Valeria Andino Célleri  <https://orcid.org/0009-0009-6350-9633>
Escuela Superior Politécnica de Chimborazo (ESPOCH)
valeria.andino@esPOCH.ec
- 2 Jimena Catalina Viteri Ojeda  <https://orcid.org/0000-0001-7049-8981>
Escuela Superior Politécnica de Chimborazo (ESPOCH)
jcviteri@esPOCH.edu.ec
- 3 Carmita Efigenia Andrade Álvarez  <https://orcid.org/0000-0003-4507-425X>
Escuela Superior Politécnica de Chimborazo (ESPOCH)
candrade_a@esPOCH.edu.ec
- 4 Verónica Janeth Argüello Pazmiño  <https://orcid.org/0000-0002-5508-9538>
Universidad Estatal de Bolívar
veronica.arguello@ueb.edu.ec



Artículo de Investigación Científica y Tecnológica

Enviado: 11/06/2023

Revisado: 27/07/2023

Aceptado: 01/08/2023

Publicado: 19/08/2023

DOI: <https://doi.org/10.33262/ap.v5i3.1.384>

Cítese:

Andino Célleri, L. V., Viteri Ojeda, J. C., Andrade Álvarez, C. E., & Argüello Pazmiño, V. J. (2023). La importancia de las matemáticas en la estructura de datos: optimización y eficiencia. AlfaPublicaciones, 5(3.1), 47–59. <https://doi.org/10.33262/ap.v5i3.1.384>



ALFA PUBLICACIONES, es una Revista Multidisciplinar, **Trimestral**, que se publicará en soporte electrónico tiene como **misión** contribuir a la formación de profesionales competentes con visión humanística y crítica que sean capaces de exponer sus resultados investigativos y científicos en la misma medida que se promueva mediante su intervención cambios positivos en la sociedad. <https://alfapublicaciones.com>

La revista es editada por la Editorial Ciencia Digital (Editorial de prestigio registrada en la Cámara Ecuatoriana de Libro con No de Afiliación 663) www.celibro.org.ec



Esta revista está protegida bajo una licencia Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International. Copia de la licencia: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

Palabras

claves:

matemáticas,
estructura de
datos,
informática,
programación.

Keywords:

mathematics,
data structure,
computer
science,
programming.

Resumen

Introducción: en este artículo, se explora la importancia de las matemáticas en la estructura de datos, un concepto fundamental en el campo de la informática. Las estructuras de datos se utilizan para organizar y almacenar información de manera eficiente, y las matemáticas proporcionan las bases teóricas necesarias para su diseño y optimización. A través de ejemplos concretos y casos de estudio, se muestra cómo los conceptos matemáticos, la teoría de conjuntos, la combinatoria y la probabilidad, son esenciales para desarrollar estructuras de datos eficientes y resolver problemas complejos. **Objetivos:** este artículo destaca la relación inseparable entre las matemáticas y la estructura de datos, y cómo su dominio es fundamental para lograr una programación eficiente y optimizada. **Metodología:** El presente artículo se basa en la investigación cualitativa a través de la revisión bibliográfica de documentos que avalan la veracidad de la información extendida en cada uno de ellos, la recopilación de los datos se enfoca en la búsqueda de artículos científicos en revistas como SciElo, Elseiver Dialnet, Skopein, entre otras, libros del área de la matemática. **Conclusiones:** las matemáticas desempeñan un papel fundamental en la estructura de datos en el campo de la informática, las estructuras de datos eficientes y optimizadas son esenciales para el desarrollo de programas informáticos de alto rendimiento. **Área de estudio general:** Matemáticas. **Área de estudio específica:** Programación e informática.

Abstract

Introduction: This article explores the importance of mathematics in data structure, a fundamental concept in the field of computer science. Data structures are used to organize and store information efficiently, and mathematics provides the theoretical foundations necessary for their design and optimization. Through concrete examples and case studies, it is shown how mathematical concepts, set theory, combinatorics, and probability, are essential to develop efficient data structures and solve complex problems. **Objectives:** This article highlights the inseparable relationship between mathematics and data structure, and how its mastery is fundamental to achieve efficient and optimized programming. **Methodology:** This article is based on qualitative research through the bibliographic review of documents that support the veracity of the

information extended in each of them, the collection of data focuses on the search for scientific articles in journals such as SciELO, Elsevier Dialnet, Skopein, among others, books around mathematics. **Conclusions:** mathematics plays a fundamental role in data structure in the field of computer science, efficient and optimized data structures are essential for the development of high-performance software. **General area of study:** Mathematics. **Specific area of study:** Programming and computer science.

Introducción

La estructura de datos es un concepto clave en el campo de la informática, ya que permite organizar y manipular la información de manera eficiente. Desde listas y árboles hasta gráficos y tablas hash, las estructuras de datos se utilizan en una amplia gama de aplicaciones informáticas. Sin embargo, para diseñar y optimizar estas estructuras de datos, es necesario comprender los conceptos matemáticos subyacentes. Las matemáticas proporcionaron las herramientas y técnicas necesarias para analizar la eficiencia de las estructuras de datos, resolver problemas de combinación y permutación, y modelar la probabilidad de ocurrencia de eventos (Cairó & Guardati, 2002). Este artículo explora en detalle la importancia de las matemáticas en la estructura de datos y cómo su aplicación adecuada puede mejorar la eficiencia y el rendimiento de los programas informáticos (Salat, 2013).

El matemático Charles Babbage creó su máquina analítica en 1834. Esta computadora tenía la capacidad de realizar las cuatro operaciones aritméticas básicas, como tener una unidad de memoria, ser programable, lo que permitía el direccionamiento condicional y los ciclos, introducir datos con tarjetas perforadas e imprimir los resultados. Por lo tanto, tenía las características de las computadoras modernas. Lamentablemente, la construcción nunca se llevó a cabo, sin embargo, el concepto de Charles Babbage tuvo un impacto significativo en el desarrollo de la computación. El matemático John Von Neumann diseñó una computadora electrónica llamada EDVAC en 1945. Fue construida y puesta en marcha en 1952. Por ejemplo, la EDVAC tenía la capacidad de resolver ecuaciones diferenciales parciales no lineales y su diseño era similar al de la mayoría de las computadoras modernas. La diferencia más significativa entre la EDVAC y las computadoras anteriores podría ser que la EDVAC podía cambiar de tarea si se introducía un programa en la misma memoria de la máquina, mientras que las computadoras anteriores solo podían realizar tareas específicas y requerían cambios de conexiones en los circuitos. La EDVAC fue la materialización de los conceptos de Charles Babbage y

una invención genial de von Neumann. La obra de von Neumann es una manera de aprender más sobre la relación entre el desarrollo de la matemática y la computación en sus orígenes (Glim et al., 1988). Stanislaw Ulam y John von Neumann realizaron una de las primeras simulaciones significativas utilizando la computadora en el estudio del fenómeno de la fisión nuclear, la difusión de neutrones (Eckhardt, 1987).

Al principio, las computadoras se programaban en lenguaje de máquina y las instrucciones se enviaban en forma de números binarios, y los programas se presentaban como listas de números binarios. La programación era un trabajo muy aburrido y con muchas posibilidades de error. Los primeros lenguajes ensamblador se desarrollaron para traducir las instrucciones escritas con nombres cortos para operaciones y números en hexadecimal al lenguaje de máquina. Después, se desarrollaron lenguajes que facilitaban la creación de algoritmos para resolver problemas y el compilador, el programa que traducía las instrucciones de un lenguaje a un lenguaje ensamblador. El primer compilador Fortran, cuyo significado es el sistema de traducción de fórmulas, nació en 1956. Este compilador tradujo un programa escrito de un lenguaje común a la gente a un programa ensamblador, es decir, en el lenguaje de la máquina. Muchos científicos pudieron utilizar la computadora en sus investigaciones y crear sus propios programas con los primeros compiladores Fortran. Sin embargo, su utilización era limitada debido a que las computadoras solo podían ser adquiridas y mantenidas por organizaciones e instituciones públicas. En 1968, la PDP-8 era una de las computadoras más baratas disponibles en la Escuela Superior de Física y Matemáticas del Instituto Politécnico Nacional. El BASIC, cuyas siglas en español son Código Simbólico de Instrucciones de Propósito General para Principiantes, fue desarrollado por Kemmeny & Kurtz (1968).

Las primeras computadoras personales aparecieron en la década de 1970, lo que permitió que cada persona tuviera una. El lenguaje Logo fue creado por Bolt et al. (1967), marcando una etapa importante de influencia en la educación. El logotipo se utiliza en la enseñanza para diseñar actividades que exploran conceptos matemáticos a través de la programación. Muchos planes de estudio incluían actividades con el lenguaje Logo.

Metodología

El presente artículo se basa en la investigación cualitativa a través de la revisión bibliográfica de documentos que avalan la veracidad de la información extendida en cada uno de ellos, la recopilación de los datos se enfoca en la búsqueda de artículos científicos en revistas como SciElo, Elseiver Dialnet, Skopein, entre otras, libros del área de la matemática en la estructura de datos y especialistas en el área, los mismos que fueron el eje principal en el desarrollo de la investigación, dando un alcance descriptivo a la investigación .

El proceso de la investigación conlleva al análisis teórico con el fin de seleccionar y correlacionar adecuadamente la información afín con el tema tratado y la misma que permita la interpretación apropiada del argumento.

Discusión

La relación entre la programación y las matemáticas

El aprendizaje en todas las áreas de la matemática tiene básicamente dos métodos debido al aumento de aplicaciones de software y plataformas educativas virtuales. El primero se centra en demostraciones y formulismo matemático que no contribuyen a la construcción de conocimientos, mientras que el segundo se centra en la enseñanza de procedimientos algorítmicos para resolver problemas tradicionales (Prieto & Cabrera, 2020; Vargas et al., 2017; Fonseca & Alfaro, 2018).

En realidad, muchas disciplinas, como la programación y la informática, se basan en las matemáticas. En términos técnicos, las matemáticas son el campo de la abstracción, es capaz de mapear situaciones de la vida real y convertirlas en números para trabajar con ellos, y luego volver a la vida real para mostrar los resultados. La programación se originó por personas que querían automatizar los cálculos.

El mundo de la programación está lleno de funciones, algoritmos, formulas y otros elementos matemáticos. Las matemáticas y la programación van de la mano, desde operaciones básicas hasta algoritmos y modelos matemáticos complejos (Joyanes, 2006).

Es un hecho que las personas que se dedican a las matemáticas tienen mucha facilidad para programar, así como las personas que se dedican a las matemáticas de igual forma. Por ello es necesario tener habilidades matemáticas para trabajar como programador.

¿No es necesario tener una comprensión completa de las matemáticas para ser un programador, pero si tiene una facilidad en esta área, es útil, si quieres ser un gran programador, tendrás que crear tus algoritmos o modelos matemáticos, realmente, los mejores programadores del mundo poseen una gran comprensión matemática como, por ejemplo, un Rovers que se envían al planeta Marte están programados por programadores muy hábiles estos utilizan matemáticas para calcular sus trayectorias?

La programación de computadoras para realizar tareas matemáticas retadoras puede mejorar la comprensión de los estudiantes "programadores" de las matemáticas relacionadas con una solución. Esto implica permitir la programación en el estudio de las matemáticas, pero enfocándose en los problemas matemáticos y en el uso del computador como herramienta para solucionar problemas de esta área (Wilson et al., 1993).

Beneficios de la programación con matemáticas

Una persona que sabe programar con matemáticas tiene las siguientes cinco ventajas:

1. Conocimientos de las funciones matemática
2. La visualización de los conceptos abstractos.
3. Son más creativos al resolver problemas complejos, resuelven problemas en menos tiempo y con menos líneas de código.
4. Aprenden rápidamente nuevos lenguajes de programación.
5. Son más capaces de razonar y analizar lógicamente.

Como ejemplo podemos hablar del Método de burbuja en JABA, es un algoritmo, funciona de manera bastante simple. recorre los elementos del arreglo y compara si un elemento es más grande o pequeño que el otro, dependiendo de si queremos ordenar de forma ascendente o descendente. Luego, hace las cantidades de vueltas necesarias hasta tener todos los elementos ordenados.

Aplicaciones de matemáticas en programación

Las matemáticas y la programación tienen muchas aplicaciones juntas. Las siguientes son algunas de las más significativas, por ejemplo:

- Criptografía: Para fines de seguridad, los algoritmos criptográficos transforman los datos en valores encriptados mediante el uso de algoritmos matemáticos.
- Minería de datos, obtienen la información correcta utilizando modelos matemáticos.
- Las criptomonedas, como el bitcoin o la cadena de bloques, son conocidas por su uso de algoritmos matemáticos complejos.
- Matrices, árboles y gráficos, algoritmos matemáticos se utilizan en tu red social favorita para sugerirte contenido relacionado a ti.

Estos son solo algunas de las aplicaciones de la programación y las matemáticas para trabajar en equipo, aunque las matemáticas están presentes en casi todos los campos, las matemáticas siempre están presentes en nuestro arte de la programación, aunque solo en pequeñas cantidades.

En qué medida las matemáticas se utilizan en la informática

En esencia, las computadoras son máquinas de cálculo que procesa datos a grandes velocidades. Cuántas matemáticas necesitas saber para tener éxito en una carrera en informática, la respuesta depende de lo avanzado que quieras ser en tu carrera de informática y de lo que quieras hacer con ella.

Las tecnologías de la información y las comunicaciones han matizado el conjunto de recursos, procedimientos y técnicas utilizados en el procesamiento, almacenamiento y transmisión de información (Menéndez et al., 2017, p. 6).

¿Las matemáticas para trabajar en informática?

Todo programador e informático debe tener un conocimiento básico de matemáticas porque las matemáticas son una parte importante de los sistemas informáticos. El tipo y el nivel de matemáticas que necesitas dependen de las áreas de la informática en las que quieras trabajar.

Algunas carreras de informática solo requieren un mínimo de habilidades matemáticas. Por ejemplo, puede crear aplicaciones móviles básicas si domina lenguajes de programación y aritmética básicos.

Pero necesitarás entender algunos conceptos matemáticos de nivel superior si quieres ir más allá en tu carrera de informática, quizás explorando campos avanzados como la seguridad informática o los sistemas operativos. Aunque nadie espera que los informáticos sean matemáticos, estudiar matemáticas en la escuela secundaria y la universidad puede ser beneficioso.

¿Cuáles son los tipos de matemáticas que se emplean en la informática?

Los informáticos emplean una variedad de matemáticas, los que trabajan con algoritmos matrices avanzados, como los que utilizan imágenes para diagnósticos médicos, deben comprender conceptos complejos como las ecuaciones diferenciales.

Conocer y dominar la teoría de los números y las ecuaciones básicas es importantes para los estudiantes del área informática, algunos temas que puedes explorar.

Contadores binarios y hexadecimales

El sistema numérico binario representa todos los conceptos matemáticos con solo dos dígitos, 0 y 1. Porque las computadoras utilizan el sistema binario en lugar del sistema decimal, que es más común, los informáticos deben estar familiarizados con él.

Aunque los computadores pueden leer en binario, puede ser incómodo para las personas. La programación informática ha utilizado el sistema hexadecimal, que consiste en nueve dígitos decimales y letras que van de la A a la F. Se convierte en binario electrónicamente y tiene más sentido para los humanos.

Ciencias matemáticas discretas

Una matemática tangible es la matemática discreta. Analiza cómo las cosas diferentes se relacionan entre sí. Las matemáticas discretas comprenden:

- Probabilidades
- Combinación de datos
- La lógica
- La teoría del número
- La teoría de los grafos

La ingeniería informática moderna se basa en los procesos de resolución de problemas de las matemáticas discretas. Impulsan disciplinas simples y complejas, como:

- Arquitectura de computadoras
- Programación basada en funciones
- sistemas dispersos
- Autoaprendizaje e inteligencia artificial

Podrías probar varios algoritmos y determinar cuál es el más eficiente utilizando las matemáticas discretas como profesional de la informática. El pensamiento lógico que desarrolles como estudiante de matemáticas discretas, por otro lado, te ayudará a abordar las tareas de programación y desarrollo de manera más metódica.

Estadísticas

En los últimos años, desde el aprendizaje automático hasta el rendimiento personalizado de las aplicaciones, los datos se han convertido en una parte cada vez más importante de la informática científica. Todos los estudiantes de informática deben estudiar estadística debido a la mayor dependencia de los datos.

Los programas informáticos pueden hacer generalizaciones y predicciones precisas basadas en la información disponible gracias a la estadística. La comprensión de la estadística puede ayudar a comprender conceptos como:

- Minería de datos
- Aprendizaje asistido
- Modelización futura
- Reconocimiento vocal
- La capacidad del usuario para responder
- Analizar una infografía

Cuanto más aprendas sobre estadística, más experto serás en administrar y usar estructuras de datos complejas. Esto le permitirá desarrollar algoritmos más complejos y aumentar la personalización del software.

Álgebra lineal

Los algoritmos son la base de la informática y la ingeniería de software. Un algoritmo es una fórmula que crea la salida adecuada a partir de una entrada específica. Si esto suena familiar, puede recordarte el álgebra de la secundaria o el instituto.

El álgebra comprende conceptos como:

- Los exponentes polinómicos son
- Cálculos lineales
- La ecuación cuadrática

Cada uno de estos conceptos tiene usos en la informática. La comprensión de las relaciones exponenciales puede ayudarte a transformar las imágenes en atractivas. El análisis de algoritmos se beneficia de la resolución de ecuaciones lineales porque permite identificar errores.

Sin saber mucho sobre álgebra, puedes desarrollarte en el desarrollo de software, pero aprenderla puede ayudarte a encontrar nuevas oportunidades.

Calcular

Es probable que tomes un curso de cálculo si tienes la intención de obtener un título en informática.

El cálculo integral y el cálculo diferencial son dos tipos diferentes de cálculo. Juntos, estas disciplinas te ayudan a calcular la tasa de cambio, que es una parte importante de muchos algoritmos y programas.

Las ecuaciones diferenciales son particularmente cruciales. Una ecuación diferencial calcula cómo cambia algo y a qué velocidad, para que una computadora pueda predecir los resultados futuros (Newman, 2008). Esto contribuye a:

- Construir simulaciones
- Mejorar los programas de resolución de problemas
- Creación de algoritmos que funcionen bien
- Crear gráficos precisos

¿Qué es necesario para comenzar en matemáticas e informática?

La comprensión de la informática mejorará con el estudio de matemáticas, los profesores de informática de hoy en día están conscientes de que todos tienen diferentes niveles de educación. Para brindar a los estudiantes las mejores oportunidades de éxito, incorporan

el aprendizaje de matemáticas a su plan de estudios. Los profesionales de la informática deben tener al menos un conocimiento básico de matemáticas (Erwin, 20008).

Con lo antes expuesto se presentan algunos ejemplos de aplicaciones de la matemática en la informática:

1. Fundamentos matemáticos en la estructura de datos:

- Teoría de conjuntos y operaciones básicas
- Combinatoria y permutaciones en la organización de datos.
- Probabilidad y distribución en la modelización de eventos.

2. Análisis de eficiencia de las estructuras de datos:

- Notación de O grande y análisis asintótico.
- Análisis de complejidad temporal y espacial.
- Optimización y selección de estructuras de datos basados en análisis matemáticos.

3. Estructuras de datos y algoritmos matemáticos:

- Listas y árboles en la representación de relaciones matemáticas.
- Gráficos y algoritmos de búsqueda en problemas de optimización.
- Tablas hash y estructuras probabilísticas en la gestión de datos.

4. Aplicaciones de las matemáticas en la estructura de datos:

- Búsqueda y ordenamiento eficiente de datos.
- Almacenamiento y recuperación de información en bases de datos.
- Procesamiento y análisis de grandes volúmenes de datos.

Conclusión

- En conclusión, las matemáticas desempeñan un papel fundamental en la estructura de datos en el campo de la informática. Las estructuras de datos eficientes y optimizadas son esenciales para el desarrollo de programas informáticos de alto rendimiento. Las matemáticas proporcionaron las bases teóricas necesarias para diseñar y analizar la eficiencia de estas estructuras, así como para resolver problemas complejos relacionados con la organización y manipulación de datos. Desde la teoría de conjuntos y la combinatoria hasta la probabilidad y el análisis de complejidad, los conceptos matemáticos son esenciales para el diseño y optimización de estructuras de datos. Dominar las matemáticas es fundamental para aquellos interesados en la estructura de datos, ya que proporciona las herramientas necesarias para desarrollar algoritmos eficientes y resolver problemas complejos de manera efectiva. En resumen, la relación entre las

matemáticas y la estructura de datos es inseparable, y aquellos que entienden y aplican los principios matemáticos en la programación tienen una ventaja significativa en el campo de la informática.

Referencia Bibliográfica

- Cairó, O. & Guardati, S. (2002). Estructuras de Datos, 2da. Edición. McGraw-Hill.
<https://es.slideshare.net/r3n3sm33/estructuras-de-datos-osvaldo-cairo>
- Eckhardt, R. (1987). Stan Ulam, John Von Neuman, and the Monte Carlo Method. *Los Alamos Science*, 15 (Special Issue). https://edisciplinas.usp.br/pluginfile.php/4946238/mod_folder/content/0/Eckhardt_LAS_1987.pdf?forcedownload=1
- Erwin Diewert, W. (2008). Cost functions, *The New Palgrave Dictionary of Economics*, 2nd Edition.
https://www.researchgate.net/publication/316577109_Cost_Functions
- Fonseca, J. & Alfaro, C. (2018). El cálculo diferencial e integral en una variable en la formación inicial de docentes de matemática en Costa Rica. *Revista Educación*, 42(2). <https://doi.org/10.15517/revedu.v42i2.25844>
- Joyanes, L. (2006). Programación en C++: Algoritmos, Estructuras de datos y objetos. McGraw-Hill. <https://combomix.net/wp-content/uploads/2017/03/Fundamentos-de-programaci%C3%B3n-4ta-Edici%C3%B3n-Luis-Joyanes-Aguilar-2.pdf>
- Menéndez, J. Y., Galeas, E. E., & Avilez, R. M. (2017). Las tecnologías de la información y las comunicaciones y su impacto en el rendimiento financiero de las empresas. Apuntes didácticos. *Opuntia Brava*, 9(1), p. 6.
opuntiabrava.ult.edu.cu
- Newman, Peter. (2008). Indirect utility function, *The New Palgrave Dictionary of Economics*, 2nd Edition.
https://link.springer.com/referenceworkentry/10.1057/978-1-349-95121-5_1245-2
- Prieto, G. & Cabrera, D. (2020). Diseño y evaluación de una estrategia lúdica de aprendizaje para enseñar Simulación de Montecarlo. *Revista Espacios*, 41(13), art. 4. <https://bit.ly/2Vhvup7>
- Salat Figols, Ramón Sebastián. (2013). La enseñanza de las matemáticas y la tecnología. *Innovación educativa (México, DF)*, 13(62), 61-74.

http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-26732013000200005&lng=es&tlng=es.)

Vargas, V., Escalante, C. & Carmona, G. (2017). Competencias matemáticas a través de la implementación de actividades provocadoras de modelos. *Educación Matemática*, 30(1), 213-236. <https://bit.ly/2Vie0ZQ>

Wilson, J. W., Fernández, M. L., & Hathaway, N. (1993). Mathematical Problem Solving. The University of Georgia – Mathematics Education, <http://jwilson.coe.uga.edu/emt725/PSsyn/Pssyn.html>].

Conflicto de intereses

No existe conflicto de intereses en relación con el artículo presentado.

El artículo que se publica es de exclusiva responsabilidad de los autores y no necesariamente reflejan el pensamiento de la **Revista Alfa Publicaciones**.



El artículo queda en propiedad de la revista y, por tanto, su publicación parcial y/o total en otro medio tiene que ser autorizado por el director de la **Revista Alfa Publicaciones**.



Indexaciones

