

Algoritmos para la resolución de problemas

Algorithms for problem solving

Iván H. Pérez-Tavera ^a

Abstract:

Is a topic that addresses the application of systematic and well-defined steps to solve computational problems. An algorithm is a set of step-by-step instructions that guide the solution of a specific problem, from inputting data to obtaining the desired result. These algorithms can vary in complexity and efficiency depending on the problem being addressed. The process of algorithm design involves identifying the problem, developing a solution strategy, writing the algorithm in a programming language, and then testing and optimizing the algorithm as needed.

Keywords:

algorithms, problem solving, systematic, step-by-step instructions, computational problems, complexity, efficiency, solution strategy, programming language, testing, optimization

Resumen:

Es un tema que aborda la aplicación de pasos sistemáticos y bien definidos para resolver problemas computacionales. Un algoritmo es un conjunto de instrucciones paso a paso que guían la solución de un problema específico, desde la entrada de datos hasta la obtención del resultado deseado. Estos algoritmos pueden variar en complejidad y eficiencia dependiendo del problema que se esté abordando. El proceso de diseño de algoritmos implica identificar el problema, desarrollar una estrategia de solución, escribir el algoritmo en un lenguaje de programación y luego probar y optimizar el algoritmo según sea necesario.

Palabras Clave:

algoritmos, resolución de problemas, sistemático, instrucciones paso a paso, problemas computacionales, complejidad, eficiencia, estrategia de solución, lenguaje de programación, prueba, optimización

Introducción

Los algoritmos son fundamentales para la resolución de problemas en diversas áreas como la programación, las matemáticas, la ingeniería y la ciencia¹. Un algoritmo es una serie de pasos lógicos y precisos que se realizan para alcanzar un objetivo específico.

La resolución de problemas mediante algoritmos generalmente sigue las siguientes fases:

Análisis del problema: Comprender la necesidad o entender el problema.

Diseño del algoritmo: Generar los pasos necesarios que describan cómo, a partir de los datos de entrada, se formarán los cálculos u operaciones necesarios para obtener los datos de salida (solución del problema).

Codificación: Implementar el algoritmo en un lenguaje de programación específico.

Compilación y ejecución: Traducir el código fuente a un formato que la máquina pueda entender y ejecutarlo.

Verificación y depuración: Probar el programa y corregir cualquier error que pueda surgir.

Algoritmos para la resolución de problemas

En el contexto de la programación y la ingeniería de software, un algoritmo es la especificación de una secuencia de pasos u operaciones que, al ser ejecutadas, permiten resolver un problema. Estos pasos deben estar expresados con operaciones comprensibles para quien las ejecutará, a las cuales llamamos primitivas.

^a Iván Horacio Pérez Tavera, Universidad Autónoma del Estado de Hidalgo, Escuela Preparatoria No. 4, Pachuca-Hidalgo, México,

<https://orcid.org/0000-0001-7708-6912>, Email: itavera@uaeh.edu.mx

El desarrollo de algoritmos para la resolución de problemas es un proceso que implica varias etapas.

Análisis del problema: Esta etapa implica entender el problema y establecer las necesidades de manera clara y concisa. Se identifican los datos de entrada y salida, y se define qué es lo que requiere el usuario del sistema de software.

Diseño del algoritmo: Aquí se crean los pasos que servirán para resolver el problema. Se describen los pasos necesarios que, a partir de los datos de entrada, formarán los cálculos u operaciones necesarios para obtener los datos de salida.

Codificación: En esta etapa, se implementa el algoritmo en un lenguaje de programación específico.

Pruebas y mantenimiento: Aquí se comprueban los resultados. Se realizan pruebas para validar y verificar que el algoritmo funciona correctamente y se hacen ajustes si es necesario.

El uso de algoritmos como estrategia didáctica es una forma de describir una solución a un problema, explicando los pasos como se debe proceder y encuadrada en los datos disponibles y en tiempo finito. Permite generar pensamiento lógico aplicable a cualquier circunstancia de nuestra vida.

Es importante destacar que el desarrollo de habilidades para resolver problemas de manera metodológica es crucial para el desarrollo de algoritmos como un paso anterior a la creación de programas de computadora. Por lo tanto, el estudio y la comprensión de los algoritmos son esenciales para cualquier persona que trabaje en campos relacionados con la tecnología y las matemáticas.

Algunos ejemplos que se mencionan en el sitio de Webmaster (2023) son:

1. **Problema de la Mochila:** Este algoritmo busca maximizar el valor de los objetos que puedes llevar en una mochila con capacidad limitada¹.
2. **Ordenamiento de Números:** Existen varios algoritmos, como el de burbuja, inserción y rápido, que organizan una lista de números en orden ascendente o descendente¹.
3. **Búsqueda Binaria:** Este algoritmo encuentra un elemento en una lista ordenada dividiéndola en mitades sucesivas¹.
4. **Árbol de Búsqueda Binaria:** Es una estructura de datos utilizada para buscar y ordenar datos eficientemente¹.
5. **Problema del Camino más Corto:** Este algoritmo encuentra la ruta más corta entre dos puntos en un grafo¹.
6. **Problema del Flujo Máximo:** Este algoritmo optimiza el flujo de un recurso a través de una red¹.
7. **Problema de la Coloración de Grafos:** Este algoritmo asigna colores a vértices de un grafo de manera que vértices adyacentes tengan colores diferentes¹.
8. **Problema del Sudoku:** Este algoritmo rellena una cuadrícula de 9x9 con números de manera que no se repitan en filas, columnas y subgrids¹.
9. **Problema del Viajante de Comercio:** Este algoritmo encuentra la ruta más corta para visitar un conjunto de ciudades una sola vez y regresar al punto de partida¹.
10. **Problema de las Torres de Hanoi:** Este algoritmo mueve una pila de discos de un poste a otro siguiendo ciertas reglas

Referencias

- [1] 1.3 Metodología para la Resolución de Problemas |. (s. f.). http://cidecama.uaeh.edu.mx/lcc/mapa/PROYECTO/libro4/13_metodologia_para_la_resolucion_de_problemas.html
- [2] Admin_Edgar. (2017, 23 junio). 1. Algoritmos || Fases de la resolución de un problema. PROGRAMMING+PROGRAMACION+ECONOMICS+55+%1. <https://aprenderaprogramar.pro/1-algoritmos-fases-de-la-resolucion-de-un-problema/>
- [3] Gálvez, J. A. S. (s. f.). Solución de problemas y algoritmos. Unidades de Apoyo Para el Aprendizaje - CUAIEED - UNAM. <https://uapa.cuaieed.unam.mx/sites/default/files/minisite/static/3fa4df86-0459-4c54-a569-87d7c387f449/UAPA-solucion-problemas-algoritmos/index.html>
- [4] Jiménez, A., & Jiménez, A. (2023a, julio 21). Aprende a crear un algoritmo eficiente para resolver problemas. El Blog de Python. <https://elblogpython.com/tecnologia/aprende-a-crear-un-algoritmo-eficiente-para-resolver-problemas/>
- [5] Jiménez, A., & Jiménez, A. (2023b, julio 26). La importancia de los algoritmos matemáticos para resolver problemas. El Blog de Python. <https://elblogpython.com/matematicas/la-importancia-de-los-algoritmos-matematicos-para-resolver-problemas/>
- [6] R. García, D. (2019). Resolución de problemas y algoritmos. Notas de clase. En Departamento de Ciencias E Ingeniería de la Computación Universidad Nacional del Sur. Recuperado 20 de abril de 2024, de <https://cs.uns.edu.ar/~drg/rpa/downloads/Clases%20Teoricas/RPA01-2019-2xh-Diego-%28Algoritmos-y-Computadoras%29.pdf>
- [7] Resolución de problemas algorítmicos y objetos de aprendizaje: una revisión de la literatura. (2020). Revista Iberoamericana Para la Investigación y el Desarrollo Educativo, 10(20). <https://doi.org/10.23913/ride.v10i20.630>
- [8] Webmaster, & Webmaster. (2023, 30 octubre). 20 Ejemplos de Problemas Algorítmicos: Que es, Tipos, Características, para que Sirve y Definición. SignificadosWeb.com. <https://significadosweb.com/ejemplos-de-problemas-algoritmicos-que-es-tipos-caracteristicas-para-que-sirve-definicion/>