

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: lizeth Barrios Retana

Matrícula: 372813

Maestro: Pedro Núñez Yépiz

Actividad No. : 9

Tema - Unidad : librerías en c, métodos de ordenación y búsqueda

Ensenada, Baja California a 08 de Octubre de 2022.

1. INTRODUCCIÓN

Las funciones y métodos de ordenación y búsqueda son utilizadas para organizar datos de manera eficiente y buscar información en conjuntos de datos, la ordenación se refiere a la tarea de reorganizar elementos en una secuencia de datos en un orden específico, mientras que la búsqueda se trata de localizar un elemento particular dentro de un conjunto de datos. Por otro lado, la búsqueda implica localizar un elemento particular en un conjunto de datos, y la eficacia de este proceso depende en gran medida de cómo estén organizados los datos.

2. COMPETENCIA

Se pretende que los programadores sean capaces de seleccionar y adaptar algoritmos de ordenación y búsqueda para obtener un rendimiento óptimo en términos de velocidad y eficiencia en la gestión de datos. Los algoritmos de búsqueda permiten localizar elementos específicos dentro de conjuntos de datos, acelerando la recuperación de información relevante.

3. FUNDAMENTOS

Algoritmos de Ordenación Comunes:

- Ordenamiento burbuja: Compara elementos adyacentes y los intercambia si están en el orden incorrecto. Repite este proceso hasta que no se requieran más intercambios.
- Ordenamiento por selección: Encuentra el elemento mínimo y lo coloca en la posición correcta, repitiendo este proceso para los elementos restantes.
- Ordenamiento por inserción: Construye una lista ordenada uno a uno, insertando elementos no ordenados en la posición correcta.
- Ordenamiento rápido (Quicksort): Divide el conjunto en subconjuntos más pequeños y ordena recursivamente. Es altamente eficiente en la mayoría de los casos.

Búsqueda secuencial: Implica examinar cada elemento uno por uno hasta encontrar el que se busca. Es simple pero puede ser ineficiente en conjuntos de datos grandes.

Comparación: La mayoría de los algoritmos de ordenación se basan en la comparación de elementos. Se evalúa si un elemento es mayor, menor o igual a otro, y se realizan intercambios en consecuencia.

4. PROCEDIMIENTO

- Realiza programa en C utilizando librería propia, el programa deberá tener el siguiente menú.
 - MENÚ
 1. LLENAR VECTOR
 2. LLENAR MATRIZ
 3. IMPRIMIR VECTOR
 4. IMPRIMIR MATRIZ

- 5. ORDENAR VECTOR
- 6. BUSCAR VALOR EN VECTOR
- 0. SALIR

NOTA: El programa deberá repetirse cuantas veces lo desee el usuario, Validado el menú con la función vali_num

INSTRUCCIONES

- 1. **LLENAR VECTOR** .- Llenar vector con 15 números, los números generados aleatoriamente, los números entre el rango de 100 al 200 (no repetidos)
- 2. **LLENAR MATRIZ** .- Llenar la matriz de 4x4 con con números generados aleatoriamente, números entre el rango de 1 al 16 (no repetidos)
- 3. **IMPRIMIR VECTOR** .- Imprime el vector que se envíe, donde la función recibe como parámetro el vector,tamaño, nombre del vector.
- 4. **IMPRIMIR MATRIZ**.- Imprime la matriz sin importar el tamaño de la matriz recibiendo como parámetros la matriz, la cantidad de renglones y columnas, así como nombre que se le dará a la matriz
- 5. **ORDENAR VECTOR**.- Usar función que ordene el vector por el método de ordenación de la Burbuja mejorada.
- 6. **BUSCAR VALOR EN VECTOR**.- Buscar un valor en el vector usando el método de búsqueda secuencial.
- 0. **SALIR**

5. RESULTADOS Y CONCLUSIONES

En conclusión, podemos afirmar que el conocimiento y dominio de los conceptos y algoritmos relacionados con las funciones y métodos de ordenación y búsqueda desempeñan un papel fundamental en el campo de la programación y la ciencia de datos. La comprensión de estas herramientas resulta esencial para organizar conjuntos de datos de manera eficaz y para recuperar información de forma veloz y precisa. Aquellos que adquieren competencia en este tema tienen la capacidad de mejorar el rendimiento de sus aplicaciones informáticas, resolver una amplia gama de problemas prácticos y tomar decisiones bien fundamentadas al elegir los algoritmos más apropiados según los requisitos particulares de cada proyecto.

Se muestran los resultados obtenidos con la elaboración de los códigos los cuales se encuentran en el pdf "CÓDIGO Y CAPTURA DE PANTALLAS" adjunto a este reporte.

6. ANEXOS

Se muestran los resultados obtenidos con la elaboración de los códigos los cuales se encuentran en el pdf "CÓDIGO Y CAPTURA DE PANTALLAS" adjunto a este reporte.

7. REFERENCIAS

Núñez Yepiz, P. (2022, 1 marzo). MP U3 3.3 condiciones anidadas P2. Recuperado 10 de septiembre de 2023, de https://drive.google.com/drive/folders/1vjg3acLVQljDX-Q5qSvJsg_T9rrQ5HB

Núñez Yepiz, P. (2021, 31 octubre). MP U3 3.3 condiciones anidadas P3. Recuperado 10 de septiembre de 2023, de https://drive.google.com/drive/folders/1vjg3acLVQljDX-Q5qSvJsg_T9rrQ5HB

Diseño de algoritmos y su codificación en lenguaje C

Corona, M.A. y Ancona, M.A. (2011)..

España: McGraw-Hill.

ISBN: 9786071505712

Programación estructurada a fondo:implementación de algoritmos en C

:Pearson Educación.Sznajdleder, P. A. (2017)..

Buenos Aires,Argentina: Alfaomega

Como programar en C/C++

H.M. Deitel/ P.J. Deitel

Segunda edición

Editorial: Prentice Hall.

ISBN:9688804711

Programación en C.Metodología, estructura de datos y objetos

Joyanes, L. y Zahonero, I. (2001)..

España:McGraw-Hill.

ISBN: 8448130138