



Universidad Autónoma de Baja California

Facultad de Ingeniería, Arquitectura y Diseño

Ingeniero en Computación

Asignatura:

Programación Estructurada

Actividad 11:

Funciones y metodos de ordenacion y busqueda

Estructuras y librerías

Brayan Arturo Rocha Meneses

Matricula:

371049

Ensenada Baja California 11 de Noviembre del 2023

Introducción:

En el mundo de la programación, las funciones y métodos de ordenación y búsqueda son herramientas fundamentales para el desarrollo de aplicaciones eficientes y lógicas. Estas funciones permiten organizar y recuperar datos de manera efectiva, lo que resulta esencial en proyectos de cualquier envergadura. Además, el uso de estructuras y librerías contribuye a la optimización del proceso de desarrollo al proporcionar una base sólida y reutilizable para nuestras aplicaciones.

En la actividad se hace un registro de alumnos que captura lo que son sus datos personales de manera manual y automática, esto proporcionará la curp generada para los dos casos y se vea en un formato de tabla

Competencia:

Los métodos de ordenación y búsqueda es un desafío que puede elevar nuestras habilidades de programación a un nivel superior. La idea principal es que al comprender profundamente estos métodos, podremos desarrollar programas de manera más eficiente y lógica. Esto significa que no dependemos constantemente de la búsqueda cuando nosotros entendemos problemas de ordenación o búsqueda en nuestros proyectos.

El desarrollo de librerías personalizadas también desempeña un papel importante en esta competencia. Al construir y mantener nuestras propias librerías, creamos un conjunto de herramientas que estarán disponibles en futuros proyectos. Esto no solo ahorra tiempo, sino que también garantiza que estemos utilizando soluciones optimizadas y probadas en lugar de escribir funciones fundamentales desde cero una y otra vez.

Además, el uso de archivos de texto para documentar y visualizar la salida de nuestro código es una práctica valiosa. Esto no solo nos permite mantener una documentación organizada, sino que también es esencial para la comunicación efectiva de datos y resultados a lo largo del ciclo de vida del proyecto.

Fundamentos:

Funciones y Métodos de Ordenación y Búsqueda:

Las funciones de ordenación y búsqueda son cruciales en la gestión de datos. Los algoritmos de ordenación nos permiten organizar elementos en una secuencia específica, lo que facilita la recuperación eficiente de datos. Los métodos de búsqueda, por otro lado, nos ayudan a localizar información deseada en grandes conjuntos de datos.

Los métodos de ordenación pueden ser de diversos tipos, desde el ordenamiento burbuja hasta el quicksort, y cada uno tiene sus propias ventajas y desventajas en términos de eficiencia y complejidad. La elección de un método de ordenación depende de la cantidad de datos y las restricciones de tiempo.

En cuanto a la búsqueda, se utilizan algoritmos como la búsqueda binaria o la búsqueda lineal. La búsqueda binaria es especialmente eficaz en conjuntos de datos ordenados, ya que reduce significativamente el tiempo de búsqueda al dividir repetidamente el conjunto de datos en dos mitades y comparar con el elemento central.

Estructuras y Librerías:

El uso de estructuras de datos y librerías predefinidas es esencial para una programación eficiente. Las estructuras de datos, como arrays, listas y árboles, organizan y almacenan datos de manera que se puedan acceder y manipular de forma eficiente. Las librerías, por otro lado, proporcionan un conjunto de funciones y métodos que abordan tareas comunes de programación, como entrada/salida, manipulación de cadenas, matemáticas y más.

El desarrollo y el mantenimiento de nuestras propias librerías y estructuras de datos pueden mejorar significativamente la reutilización del código y simplificar la creación de futuros programas. Además, nos brinda la flexibilidad de adaptar estas herramientas a las necesidades específicas de nuestros proyectos, lo que resulta en un código más limpio y eficiente.

Funciones:

1. **Abstracción:** Las funciones permiten abstraer la complejidad de un programa al dividirlo en partes más pequeñas y manejables. Esto facilita la comprensión y el mantenimiento del código.
2. **Reutilización de código:** Las funciones se pueden reutilizar en diferentes partes de un programa o incluso en proyectos diferentes, lo que ahorra tiempo y esfuerzo al no tener que volver a escribir el mismo código.
3. **Modularidad:** Las funciones promueven la modularidad al separar las tareas en unidades independientes. Esto hace que el código sea más fácil de depurar y mantener.
4. **Pasaje de parámetros:** Las funciones pueden aceptar parámetros, lo que les permite recibir datos de entrada y devolver resultados. Esto permite que las funciones sean flexibles y adaptables a diferentes situaciones.

Métodos de Ordenación y Búsqueda:

1. **Eficiencia:** Los métodos de ordenación y búsqueda se diseñan para realizar tareas específicas de manera eficiente. La elección de un método depende de factores como el tamaño de los datos y la velocidad requerida.
2. **Ordenación:** La ordenación es esencial para organizar datos de manera que sea más fácil recuperar la información necesaria. Los algoritmos de ordenación, como el quicksort y el mergesort, se basan en comparaciones y manipulación de elementos.
3. **Búsqueda:** Los métodos de búsqueda son esenciales para localizar datos específicos en conjuntos de información extensos. La búsqueda binaria y la búsqueda lineal son dos ejemplos comunes que se basan en diferentes estrategias para encontrar datos.
4. **Complejidad:** La complejidad de los algoritmos de ordenación y búsqueda se mide en términos de tiempo y espacio. Comprender esta complejidad es fundamental para elegir el algoritmo adecuado en función de los requisitos del proyecto.

Estructuras y Librerías:

1. **Estructuras de datos:** Las estructuras de datos como arrays, listas, pilas y colas permiten organizar y almacenar datos de manera eficiente. Cada estructura tiene ventajas y desventajas según el tipo de datos y las operaciones requeridas.
2. **Reutilización de código:** Las librerías ofrecen un conjunto de funciones predefinidas que pueden utilizarse en múltiples proyectos. Esto fomenta la reutilización de código y ahorra tiempo de desarrollo.
3. **Optimización de recursos:** Las librerías se desarrollan para ser eficientes y confiables. Al usar librerías probadas, los programadores pueden confiar en que el código base es sólido y seguro.
4. **Personalización:** La creación de librerías y estructuras personalizadas permite a los programadores adaptar herramientas a las necesidades específicas del proyecto. Esto ofrece una flexibilidad valiosa y una mayor autonomía.

Procedimiento:

PRÁCTICA 11

Realiza el programa que contenga el siguiente menú

M E N Ú

- 1.- Cargar
- 2.- Eliminar
- 3.- Buscar
- 4.- Ordenar
- 5.- Imprimir
- 6.- Archivo Texto
- 0.- Salir

El programa deberá poder almacenar en un arreglo (máximo 2,000 registros) los datos para generar el CURP **la estructura debe contener 2 estructuras anidadas**, nombre y fecha nacimiento y un campo donde se escribirá automáticamente el curp basado en los datos proporcionados

MENÚ DESCRIPCIÓN:

- 1.- **Cargar**: Se deberá agregar 100 registros en forma automática y aleatorios (cuidar no se desborde Arreglo)
 - 2.- **Eliminar**: La búsqueda se realizará por matrícula, Imprimir el registro encontrado en forma de registro y preguntar si quiere eliminar si o no. (Eliminado Lógico x campo status)
 - 3.- **Buscar**: La búsqueda se realizará por matrícula, el programa deberá ser capaz de realizar la **búsqueda secuencial o Binaria** según sea el caso. Imprimir el registro encontrado en forma de registro.
 - 4.- **Ordenar**: La ordenación será por MATRICULA usar función de ordenación más adecuada según sea el caso **usar 2 métodos de ordenación** y el programa decidirá cuál es el que usará dependiendo del estado y tamaño de registros dentro del arreglo.
Nota: (validar si el arreglo ya está ordenado no volver ordenar por el mismo campo)
 - 5.- **Imprimir**: El programa deberá imprimir los datos del arreglo (**solo registros activos**) en forma de tabla en pantallas de **40 registros** y presionando la tecla de continuar en cada uno de los casos.
 - 6.- **Archivo de Texto**: El programa deberá generar un archivo de texto con los datos del arreglo (**solo registros activos**) formatear salida.
- NOTA:** forma de registro es de la siguiente manera:

MATRICULA : 300523
NOMBRE : YAREMI
NOMBRE2 : GHIZETH
AP PATERNO : GARCIA
AP MATERNO : GUERRERO
FECHA NAC : 03-04-2010
EDAD : 19
SEXO : MUJER
LUGAR NAC : BAJA CALIFORNIA SUR
CURP : GAGY030410MBCRRRA5

NOTA : Librería Propia, Usar funciones, no se permiten variables global

Conclusiones:

Convertirse en expertos en la ordenación y búsqueda, el uso de estructuras y librerías, y la implementación de archivos de texto en nuestros proyectos son pasos importantes en la mejora de nuestras habilidades de programación. Estas habilidades nos permiten ser más eficientes, lógicos y autónomos a la hora de desarrollar aplicaciones, al tiempo que mantienen una documentación organizada y proporcionan una base sólida para futuros proyectos. En última instancia, nos convierte en programadores más capacitados y versátiles, capaces de abordar desafíos con mayor confianza y creatividad.

Anexo: