

Práctica 2

Temario:

- Algoritmos de búsqueda y ordenamiento.
- Structs y arreglos de structs.

Ejercicios:

Ejercicio 1.

Escribir un programa en C++ que realice una búsqueda secuencial en un arreglo desordenado de cadena de caracteres. El programa debe solicitar al usuario que ingrese, por teclado, nombres de diferentes colores (elegir una condición de corte apropiada). Éstos deben ser almacenados en un arreglo. Por otra parte, se debe ofrecer una opción de búsqueda, donde se solicite al usuario ingresar el nombre de un color e informar si dicho color se encuentra o no en el arreglo.

Ejercicio 2.

Escribir un programa en C++ que realice una búsqueda secuencial en un arreglo de números enteros ordenado de menor a mayor. El programa debe solicitar al usuario que ingrese, por teclado, una serie de números (elegir una condición de corte apropiada). Éstos deben ser almacenados en forma ordenada en un arreglo. Por otra parte, se debe ofrecer una opción de búsqueda, donde se solicite al usuario ingresar un número e informar si se encuentra o no en el arreglo.

Ejercicio 3.

Escribir un programa en C++ que realice una búsqueda binaria en un arreglo de números enteros donde algunos números pueden llegar a repetirse. El programa debe encontrar la primera aparición del número buscado e informarlo en pantalla.

Ejercicio 4.

En un comercio de artículos electrónicos se lleva un registro de los precios de los productos disponibles. El administrador desea ordenar estos precios de menor a mayor para facilitar su visualización. Como desarrollador, te han solicitado implementar un programa en C++ que realice esta tarea utilizando el algoritmo de ordenamiento por inserción.

Requerimientos:

- El programa debe permitir al usuario ingresar la cantidad de productos y sus respectivos precios.

PROGRAMACIÓN IMPERATIVA

Trabajo Práctico Nº 2

Escuela de Tecnología | Área Algoritmos y Lenguajes

- Se debe implementar el algoritmo de ordenamiento por inserción para ordenar los precios en orden ascendente.
- Una vez ordenados, el programa debe mostrar los precios ordenados en pantalla.

Instrucciones adicionales:

- Utilizar un arreglo para almacenar los precios.
- Implementar una función separada que realice el ordenamiento por inserción.
- La cantidad de productos y los precios deben ser ingresados por el usuario.

Ejercicio 5.

Escribir un programa en C++ que realice las siguientes tareas: **a)** generar, aleatoriamente, una lista (arreglo) con 100 números enteros en el rango de 0 a 100.000; **b)** Ordenar el listado en modo creciente por el método de la burbuja; **c)** Ordenar el listado en modo decreciente por el método de la burbuja.

Ejercicio 6.

Considerando la siguiente definición de struct:

```
struct TipoZapato {  
    string estilo;  
    double precio;  
};
```

¿Qué salidas producirá el siguiente código?

```
TipoZapato zapato1, zapato2;  
zapato1.estilo = "Sandalia";  
zapato1.precio = 9.99;  
cout << zapato1.estilo << " $" << zapato1.precio << endl;  
zapato2 = zapato1;  
zapato2.precio = zapato2.precio / 9;  
cout << zapato2.estilo << " $" << zapato2.precio << endl;
```

Ejercicio 7.

Realizar un algoritmo que almacene información de, como máximo, 500 libros en un arreglo estático. Un libro se define como un struct con los siguientes campos: titulo, autor, ISBN, editorial, cantidadHojas. El algoritmo finaliza luego de cargar el libro "El hobbit", el cual debe procesarse o cuando ya no quede espacio en el arreglo (lo que suceda primero). Finalmente, imprimir el arreglo resultante mostrando todos los datos de cada libro.

PROGRAMACIÓN IMPERATIVA

Trabajo Práctico Nº 2

Escuela de Tecnología | Área Algoritmos y Lenguajes

Ejercicio 8.

Realizar un programa que almacene en un arreglo la lista de invitados a una fiesta. El salón donde se realizará, tiene capacidad máxima para 150 personas.

De cada invitado se conoce su nombre, apellido, sexo y edad. Se ingresarán datos hasta que el nombre sea igual a "ZZZ" o hasta que la capacidad del salón esté completa.

Una vez finalizada la carga:

- A. Informar la cantidad de invitados que asistirán a la fiesta.
- B. Realizar una función que liste los nombres de las mujeres cuya edad sea ≤ 20 .
- C. Realizar una función que, dado un apellido y nombre, permita eliminar un invitado en caso de no poder asistir.

Ejercicio 9.

Escribir un programa que almacene en un arreglo los datos de los alumnos de una escuela luego de finalizado el año académico. Como máximo, puede haber 5.000 alumnos. De cada alumno se debe guardar el nombre y la calificación obtenida en el primer, segundo y tercer trimestre del año académico. Imprimir el listado de nombres de alumnos mostrando un mensaje de "APTO" si el alumno supera o iguala la calificación de 5 para todas las notas o "NO APTO" si no lo alcanza. Utilizar un arreglo como campo del struct para almacenar las tres calificaciones.

Ejercicio 10.

La universidad posee información histórica sobre los estudiantes (no más de 1.000) de una materia determinada. De cada uno almacena: nombre y apellido, legajo, cantidad de inasistencias a clase, calificación obtenida. Almacenar esta información en un arreglo de structs. Se solicita generar los siguientes reportes:

- a) Imprimir nombre y apellido de los alumnos que tuvieron más de 5 inasistencias.
- b) Imprimir número de legajo de los alumnos cuya calificación promedio sea mayor o igual a la calificación promedio total (requiere calcular un promedio de las calificaciones de todos los alumnos en el listado).
- c) Imprimir el número de legajo de aquellos alumnos que tienen promedio mayor o igual a 9.
- d) Dado el legajo de un alumno, eliminarlo del listado.

Nota: analizar si es posible reutilizar código en más de uno de los incisos.

Ejercicio 11.

Implementar un programa que almacene en un arreglo estático, los datos de los ciudadanos que solicitaron un certificado de antecedentes en el Registro Nacional de las Personas. De cada ciudadano se conoce: nombre y apellido, fecha de nacimiento (tipo long: formato AAAAMMDD), dirección, número de documento, tipo de documento (tipo int: 1 -DNI-, 2 -Pasaporte-, 3 -Otro-) y sexo (tipo char: 'M', 'F').

- a) Hacer un módulo de carga del arreglo, utilizando una condición de corte adecuada. Se debe validar el tipo de documento (1, 2 ó 3) y dejar al usuario en un bucle mientras ingrese un número inválido.
- b) Imprimir el listado, con el siguiente formato de ejemplo:
Nombre y apellido: Juan Perez
Dirección: Levalle 132
Fecha de nacimiento: 27/10/1979
Tipo de documento: DNI
Número de documento: 27439221
Sexo: masculino
 - La fecha debe mostrarse en formato dd/mm/aa. El tipo de documento debe mostrarse con su nombre.
 - El género debe mostrar la palabra completa.
- c) Implementar un módulo que, dado el arreglo original, cargue en otro arreglo los ciudadanos de sexo masculino únicamente. Imprimir este nuevo arreglo.
- d) Hacer lo mismo que en el punto anterior, pero obteniendo sólo las mujeres. Modularizar para no repetir código.

Ejercicio 12.

Escribir un programa para gestionar las sucursales de una empresa.

- a) Almacenar en un arreglo (máximo 2.000 elementos) los datos de los empleados: legajo, nombre y DNI. En otro arreglo (máximo 20 elementos) almacenar la información de las sucursales: nombre de la sucursal y legajo del encargado. Cada vez que se ingrese un legajo del encargado de una sucursal, se debe validar si corresponde a un empleado existente. Si no existe, dejar al usuario en un bucle hasta que ingrese un número de legajo existente.
- b) Dado el legajo de un empleado, informar su nombre.
- c) Imprimir un listado de sucursales, mostrando nombre de sucursal y nombre del empleado encargado.

Nota: analizar si es posible reutilizar código en más de uno de los incisos.

Ejercicio 13.

Implementar un sistema que permita administrar datos de clientes de un comercio para enviarles promociones. De cada cliente se tiene: nombre y apellido, DNI, e-mail, fecha de cumpleaños en formato DDMM, monto total de compras en el último año.

- A. Realizar el módulo de carga. La misma finaliza con el nombre "z". Los clientes deben almacenarse ordenados por número de DNI.
- B. Informar cuántos clientes cumplen años en cada uno de los 12 meses del año. Utilizar un arreglo auxiliar para contabilizarlos.
- C. Informar el nombre del cliente con el mayor monto de compras.