

Enunciado

Título: Gestión de Clientes en Manejo de estructuras

Descripción del Problema:

Te encargamos un proyecto en el cual estas encargado de desarrollar un programa en C que permita gestionar los datos de clientes para una pequeña empresa. Los clientes deben estar organizados de manera que la empresa pueda consultarlos por su código, (**podrías armar un lindo menú**).

Para ello, deberás implementar un sistema que permita:

1. Tener un menú de ingreso para poder seleccionar lo que el usuario desee realizar.
2. Cargar los datos de varios clientes, eso lo tienen que limitar a **5** con **MAX_CLI** clientes así no pierden el tiempo con la carga, incluyendo su **código, nombre, apellido y fecha de nacimiento**.
3. Ordenar los clientes en función de su código (**de menor a mayor**).
4. Ordenar los clientes en función de su apellido (**alfabéticamente**).
5. Mostrar los datos de los clientes ordenados según los criterios anteriores.

Estructuras de Datos Para Utilizar:

- **Fecha:** Una estructura que almacena el **día, mes y año** de nacimiento del cliente.
- **Cliente:** Una estructura que almacena el **código, nombre, apellido y fecha** de nacimiento de un cliente.

```
• #include <stdio.h>
• #include <string.h>
• #define MAX_CLI 3
• typedef struct
• {
•     int dia;
•     int mes;
•     int anio;
• }Fecha;
•
• typedef struct
• {
•     int codigo;
•     char nombre[50];
•     char apellido[50];
•     Fecha fechaNacimiento;
• }Cliente;
```

Funciones que Debes Implementar:

1. **void cargarCliente(Cientecliente):** Esta función se encargará de cargar los datos de un cliente, solicitándolos al usuario. Debes asegurarte de que el usuario ingrese correctamente todos los campos y que el código ingresado no se repita dentro de nuestra estructura.)
2. **void mostrarCliente(Ciente[], char *, int):** Esta función mostrará en pantalla los datos de los clientes, incluyendo su código, nombre, apellido y fecha de nacimiento.
3. **void ordenarClientesPorCodigo(Ciente[], int):** Implementar esta función para ordenar los clientes por su código utilizando el algoritmo de burbuja.
4. **void ordenarClientesPorApellido(Ciente[], int):** Implementa esta función para ordenar los clientes por su apellido, también utilizando el algoritmo de burbuja, pero comparando cadenas de caracteres.
5. **void mostrarClientePorCodigo(Ciente clientes[], int, int):** Esta función muestra el cliente cargado con sus datos, se pasa por parámetro la estructura del cliente, la cantidad total de clientes y el código de cliente a buscar.
6. **int buscarCliente(int, Ciente[], int):** Implementar esta función para buscar un cliente y determinar si existe o no en nuestra estructura en el momento del alta del cliente.

Hay que recordar que la cantidad que figura en las funciones no es la máxima cantidad ingresada.

Requisitos Adicionales:

- El programa debe permitir la carga de hasta 5 clientes como máximo, tipo una versión trial.
- Asegúrate de utilizar la función **stricmp** para la comparación de apellidos, la cual permite comparar cadenas de caracteres sin tener en cuenta mayúsculas y minúsculas.

Objetivo:

El objetivo de este ejercicio es que pongas en práctica la manipulación de estructuras en C, la implementación de funciones que operan sobre ellas, y el uso de algoritmos de ordenación básicos.

A

lgunas funciones adicionales que podrán necesitar

void pausa(void)

```
{  
    printf("\nPresione Enter para continuar...");  
    fflush(stdin); // aqui si va porque puede quedar el /n  
    getchar();  
}
```

int menu(void)

```
{  
    int op;  
    do  
    {  
        system("cls");  
        printf("\n\t\t\t MENU DE OPCIONES DE CLIENTES\n");  
        printf("\n\n\t\t\t - Ingresar Datos      <1>\n");  
        printf("\n\n\t\t\t Buscar un cliente   <2>\n");  
        printf("\n\n\t\t\t Listado de Cliente Ordenado porCodigo    <3>\n");  
        printf("\n\n\t\t\t Listado de Clientes Ordenado Por Apellido  <4>\n");  
        printf("\n\n\t\t\t - Salir              <5>\n");  
        printf("\n\n\t\t\t Ingrese Opcion [1-5] : ");  
        scanf("%d", &op);  
    } while (op < 1 || op > 5);  
    return op;  
}
```