



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: Antonio Ramos Gonzalez

Matrícula: 372576

Maestro: Pedro Núñez Yépiz

Actividad No. : 11

Tema - Unidad : Librerías y Métodos de Ordenación

Ensenada Baja California a 1 de noviembre del 2023



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

1. INTRODUCCIÓN

Se implementarán distintas librerías, creadas por nosotros los alumnos, para facilitar la realización de la práctica, accediendo a estas en nuestro programa principal. Dicho programa tendrá un menú de elecciones en el cual se elegirá lo que se desea realizar.

2. COMPETENCIA

Comprender el funcionamiento de los archivos, haciendo que dichos archivos lean datos enviados por el usuario. También se espera la comprensión de los distintos métodos de búsqueda y ordenación, sabiendo cuando aplicar cada uno.

Se busca reforzar los distintos conocimientos obtenidos en prácticas anteriores, como lo son las cadenas y distintas validaciones

3. FUNDAMENTOS

Los Archivos de texto son un tipo de archivo en el cual se pueden almacenar distintos tipos de datos. La manera de acceder a ellos es mediante un tipo de dato file en el cual se declarará una variable de tipo puntero

FILE *fa

Posteriormente se accederá a la ubicación del archivo y se decidirá si se leerá el archivo o si escribiremos en él.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

4. PROCEDIMIENTO

Realiza el programa que contenga el siguiente menú

M E N Ú

- 1.- Agregar
 - a) manual (1)
 - b) Automatico (100)
 - c) Regresar
- 2.- Eliminar Registro
- 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 automáticamente 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.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

5. RESULTADOS Y CONCLUSIONES

El uso de archivos es indispensable para almacenar grandes cantidades de información, así como los distintos métodos de ordenación y búsqueda son sumamente importantes para una buena optimización de código.

```
void menu()
{
    int opc, elec, kay;
    int i = 0, j, band, bus;
    Treg reg[N];
    do
    {
        opc = msg();
        switch (opc)
        {
            case 1:
                elec = msg2();
                if (elec == 1)
                {
                    if (i < N)
                    {
                        reg[i] = agregarM(reg, i);
                        i++;
                        band = 1;
                    }
                    else
                    {
                        printf("Base de datos llena\n");
                        system("PAUSE");
                    }
                }
                if (elec == 2)
                {
                    if (i < N)
                    {
                        for (j = 0; j < 100; j++)
                        {
                            if (i < N)
                            {
                                reg[i] = cargar(reg, i);
                                i++;
                                band = 1;
                            }
                        }
                    }
                    else
                    {
                        printf("Base de datos llena\n");
                        system("PAUSE");
                    }
                }
                break;
            case 2:
                kay = valid("Ingrese matricula que desea eliminar: ", 300000, 399999);
                if (band)
                {
                    bus = searchSec(reg, i, kay);
                    pintOneReg(reg, bus);
                }
                else
                {
                    bus = searchBin(reg, 0, i, kay);
                    pintOneReg(reg, bus);
                }
                if (valid("DESEA ELIMINAR ALUMNO\n1.-SI\n0.-NO\n", 0, 1))
                {
                    reg[bus].status = 0;
                }
                break;
            case 3:
                kay = valid("Ingrese matricula que desea Buscar: ", 300000, 399999);
                if (band)
                {
```

6. ANEXOS

REPOSITORIO DE GITHUB: https://github.com/Anrago/Programacion-estructurada/tree/main/Actividad_11



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

7. REFERENCIAS

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