



**Alumno:**  
**Ramírez Monjaraz Miguel Ángel**

**Profesor:**  
**Pedro Nunez Yepiz**

**Ingeniería en computación**  
**Grupo: 432**

**Materia:**  
**Programación - Estructurada**

**Tema:**  
**Actividad 10 - FUNCIONES y METODOS DE ORDENACION Y BUSQUEDA**  
**ESTRUCTURAS Y LIBRERÍAS.**  
**ANEXOS**

**Fecha de entrega:**  
**10 de Octubre del 2023**



## Código principal

```
//                               Ensenada, Baja California a 07 de
Octubre del 2023                //
// Nombre:                      Miguel Angel Ramirez Monjaraz
//
// Matricula:                   372205
//
// Programa:                    Libreria PROPIA, DEL MIGUEL, SOBRE FUNCIONES
REUTILIZABLES                   //
// Nombre del programa: MARM_ACT9_01
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include "Gauss.h"

#define N 500

typedef struct _alum
{
    int status;
    int matri;
    char nombre[30];
    char appaterP[30];
    char apmater[30];
    int sexo;
    int edad;
} Talum;

Talum gen_regmanu();
Talum gen_rega_alea();
Talum eliminar_regis(Talum vect[], int *n);
void mostra_registros(Talum vect[], int n);
void ordenar_regis(Talum vect[], int n);
void buscar_regi(Talum vect[], int n);
void letrero();
void menu();
```



```
int main()
{
    menu();
    return 0;
}

void menu()
{
    int x = 0;
    int i = 0, op;
    Talum vect[500];
    do
    {
        letrero();
        op = valid_num(0, 6, "Ingresa la accion a realizar: ");
        int j = i + 10;
        switch (op)
        {
            case 0:
                op = 0;
            case 1:
                if (i <= 490)
                {
                    for (i; i < j; i++)
                    {
                        vect[i] = gen_rega_alea();
                    }
                }
                else
                {
                    printf("Te queda solo una opción más para ingresar 10
registros automáticamente\n");
                }
                printf("%d", i);
                break;
            case 3:
                eliminar_regis(vect, &i);
                break;
            case 2:
                if (i < 500) {
                    vect[i] = gen_regmanu();
                    i++;
                } else
```



```
{
    printf("Sin espacio para continuar el registro\n");
    op = 0;
}
break;
case 4:
    buscar_regi(vect,i);
    break;
case 5:
    ordenar_regis(vect, i);
    break;
case 6:
    mostra_registros(vect, i);
    break;
default:
    break;
}
} while (op != 0);
}

void letrero()
{
    printf("
                MENU
            \n");
    printf("1 - AGREGAR (AUTOM 10 REGISTROS)\n");
    printf("2 - AGREGAR MANUAL\n");
    printf("3- ELIMINAR REGISTRO (lógico)\n");
    printf("4.- BUSCAR\n5- ORDENAR\n6.- IMPRIMIR\n0.- SALIR\n ");
}

Talun gen_regmanu()
{
    Talun regis;
    regis.status = 1;
    regis.matri = valid_num(300000, 399999, "Ingresa tu martirucla");
    printf("Ingresa tu nombre: ");
    fflush(stdin);
    gets(regis.nombre);
    printf("Ingresa tu apellido parterno: ");
    fflush(stdin);
    gets(regis.appaterP);
    printf("Ingresa tu apellido materno: ");
    fflush(stdin);
    gets(regis.apmater);
```



```
printf("Presiona: \n");
printf("1 - Hombre \n");
printf("2 - Mujer \n");
regis.sexo = valid_num(1, 2, "Ingresa tu sexo");
regis.edad = valid_num(1, 100, "Ingresa tu edad");
return regis;
}

Talum gen_rega_alea()
{
    Talum regis;

    char *nHombre[10] = {"Erick", "Juan", "Pedro", "Luis", "Carlos",
"Miguel", "Alejandro", "José", "Javier", "Antonio"};
    char *nMujer[10] = {"Maria", "Ana", "Laura", "Sofia", "Paula",
"Isabel", "Carmen", "Patricia", "Raquel", "Marta"};
    char *apPater[10] = {"Gonzalez", "Lopez", "Rodriguez", "Martinez",
"Perez", "Fernandez", "Gomez", "Sanchez", "Ramirez", "Torres"};
    char *apMater[10] = {"Flores", "Morales", "Vazquez", "Jimenez",
"Reyes", "Diaz", "Torres", "Gutierrez", "Ruiz", "Mendoza"};
    regis.status = 1;
    regis.matri = (rand() % 100000) + 300000;
    int sexo = rand() % 2;
    if (sexo == 1)
    {
        strcpy(regis.nombre, nHombre[rand() % 10]);
    }
    else
    {
        strcpy(regis.nombre, nMujer[rand() % 10]);
    }
    strcpy(regis.appaterP, apPater[rand() % 10]);
    strcpy(regis.apmater, apMater[rand() % 10]);
    regis.sexo = sexo;
    regis.edad = (rand() % 88) + 13;
    return regis;
}

Talum eliminar_regis(Talum vect[], int *n)
{
    int matri = valid_num(300000, 399999, "Ingresa la matricula del
registro que deseas borrar: ");
    int k = 0;
```



```
Talum veck[N];

for (int i = 0; i < *n; i++)
{
    if (matri != vect[i].matri)
    {
        //Va metiendo los datos que no sean de la matricula buscada
        veck[k] = vect[i];
        k++;
    }
}

if (k == *n)
{
    // Lo que hace es que si el for anterior se completo
    // completamente es porque no se borro ninguna matricula y retorna el
    // vector dado
    printf("Matrícula no encontrada\n");
    return vect[N];
}
else
{
    printf("Registro con matrícula %d eliminado.\n", matri);
    *n = k;
    // Va actualizando el registro que le dimos porque eliminamos
    // el registro
    for (int i = 0; i < k; i++)
    {
        vect[i] = veck[i];
    }
    //retorno ya el vector dado
    return vect[k];
}
}
```



```
void mostra_registros(Talum vect[], int n)
{
    printf("-----\n");
    printf("-----\n");
    printf("| N° | MATRICULA | NOMBRE | AP.PATERNO | AP. MATERNO | SEXO | EDAD |\n");
    printf("-----\n");
    for (int i = 0; i < n; i++)
    {
        printf("| %4d | %11d | %12s | %12s | %13s | %5d | %5d |\n", i,
vect[i].matri, vect[i].nombre, vect[i].appaterP, vect[i].apmater,
vect[i].sexo, vect[i].edad);
    }
    printf("-----\n");
}

void ordenar_regis(Talum vect[], int n)
{
    int i, j;
    Talum temp;
    for (i = 0; i < n - 1; i++)
    {
        for (j = i + 1; j < n; j++)
        {
            if (vect[j].matri < vect[i].matri)
            {
                temp = vect[i];
                vect[i] = vect[j];
                vect[j] = temp;
            }
        }
    }
}
```



```
void buscar_regi(Talum vect[], int n)
{
    int i;
    int matri = valid_num(300000, 399999, "Ingresa la matricula del
registro que deseas buscar: ");

    for (i = 0; i < n; i++)
    {
        if (matri == vect[i].matri)
        {
            printf("Se encontro el usuario con matricula %d en el
registro.\n", matri);

            printf("-----\n");
            printf("|  N°  |  MATRICULA  |  NOMBRE  |  AP.PATERNO  |
AP. MATERNO  |  SEXO  |  EDAD  |\n");

            printf("-----\n");
            printf("|  %4d  |  %11d  |  %12s  |  %12s  |  %13s  |  %5d  |  %5d  |\n",
i, vect[i].matri, vect[i].nombre, vect[i].appaterP, vect[i].apmater,
vect[i].sexo, vect[i].edad);

            printf("-----\n");

            return;
        }
    }
    if(i == n)
    {
        printf("No se encontro el usuario con matricula %d", matri);
    }
}
```





Salida por pantalla:

```

                                MENU
1 - AGREGAR (AUTOM 10 REGISTROS)
2 - AGREGAR MANUAL
3- ELIMINAR REGISTRO (lógico)
4.- BUSCAR
5- ORDENAR
6.- IMPRIMIR
0.- SALIR
  Ingresar la acción a realizar:
1

```

```

2
  Ingresar tu matrícula
372290
  Ingresar tu nombre: miguel angel
  Ingresar tu apellido paterno: ramirez
  Ingresar tu apellido materno: monjaraz
  Presiona:
1 - Hombre
2 - Mujer
  Ingresar tu sexo
1
  Ingresar tu edad
19

```

```

4
  Ingresar la matrícula del registro que deseas buscar:
370041
  No se encontró el usuario con matrícula 370041

```



Ingresar la acción a realizar:

6

Nº	MATRICULA	NOMBRE	AP.PATERNAL	AP. MATERNO	SEXO	EDAD
0	300041	Carlos	Gonzalez	Mendoza	1	73
1	311478	Laura	Perez	Diaz	0	86
2	323281	Juan	Lopez	Diaz	1	75
3	304827	Ana	Perez	Vazquez	0	78
4	300292	Ana	Gomez	Ruiz	0	20
5	305447	Ana	Ramirez	Mendoza	0	37
6	325667	Miguel	Perez	Jimenez	1	64
7	331322	Luis	Perez	Morales	1	68
8	328253	Patricia	Perez	Vazquez	0	34
9	320037	Luis	Lopez	Mendoza	1	87
10	372205	miguel	angel	ramirez	1	19
11	372290	miguel angel	ramirez	monjaraz	1	19

#### MENU

- 1 - AGREGAR (AUTOM 10 REGISTROS)
- 2 - AGREGAR MANUAL
- 3- ELIMINAR REGISTRO (lógico)
- 4.- BUSCAR
- 5- ORDENAR
- 6.- IMPRIMIR
- 0.- SALIR

Ingresar la acción a realizar:

5

#### MENU

- 1 - AGREGAR (AUTOM 10 REGISTROS)
- 2 - AGREGAR MANUAL
- 3- ELIMINAR REGISTRO (lógico)
- 4.- BUSCAR
- 5- ORDENAR
- 6.- IMPRIMIR
- 0.- SALIR

Ingresar la acción a realizar:

6

Nº	MATRICULA	NOMBRE	AP.PATERNAL	AP. MATERNO	SEXO	EDAD
0	300041	Carlos	Gonzalez	Mendoza	1	73
1	300292	Ana	Gomez	Ruiz	0	20
2	304827	Ana	Perez	Vazquez	0	78
3	305447	Ana	Ramirez	Mendoza	0	37
4	311478	Laura	Perez	Diaz	0	86
5	320037	Luis	Lopez	Mendoza	1	87
6	323281	Juan	Lopez	Diaz	1	75
7	325667	Miguel	Perez	Jimenez	1	64
8	328253	Patricia	Perez	Vazquez	0	34
9	331322	Luis	Perez	Morales	1	68
10	372205	miguel	angel	ramirez	1	19
11	372290	miguel angel	ramirez	monjaraz	1	19