



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño



Ingeniería en Computación

Programación Estructurara/36276

Santos Tirado Martin/00369705

Pedro Nuñez Yepiz

Actividad N. 9

funciones y librerías

Ensenada Baja California, 06 de octubre del 2023

Introducción:

Veremos ejercicios que se resolverá con funciones y librerías propias

Competencia:

El alumno llevará a cabo los ejercicios planteados con ayuda del profesor que dejo en un documento sobre el código de esqueleto, también, el alumno resolverá los ejercicios planteados por el profesor utilizando como base el código de esqueleto, matrices y vectores, funciones, librerías propias,etc.

Fundamentos:

El alumno aprenderá sobre matrices y vectores con función de esqueleto para que pueda y aprenda a manejar perfectamente las funciones void.

Procedimiento:

Realiza programa en C utilizando librería propia, el programa deberá tener el siguiente menú.

MENÚ

1.- LLENAR VECTOR

2.- LLENAR MATRIZ

3.- IMPRIMIR VECTOR

4.- IMPRIMIR MATRIZ

5.- ORDENAR VECTOR

6.- BUSCAR VALOR EN VECTOR

0.- SALIR

NOTA: El programa deberá repetirse cuantas veces lo desee el usuario, Validado el menú con la función vali_num

INSTRUCCIONES

1.- LLENAR VECTOR .- Llenar vector con 15 números, los números generados aleatoriamente, los números entre el rango de 100 al 200 (**no repetidos**)

2.- LLENAR MATRIZ .- Llenar la matriz de 4x4 con con números generados aleatoriamente, números entre el rango de 1 al 16 (**no repetidos**)

3.- IMPRIMIR VECTOR .- Imprime el vector que se envíe, donde la función recibe como parámetro el vector, tamaño, nombre del vector.

4.- IMPRIMIR MATRIZ.- Imprime la matriz sin importar el tamaño de la matriz recibiendo como parámetros la matriz, la cantidad de renglones y columnas, así como nombre que se le dará a la matriz

5.- ORDENAR VECTOR.- Usar función que ordene el vector por el método de ordenación de la Burbuja mejorada.

6.- BUSCAR VALOR EN VECTOR.- Buscar un valor en el vector usando el método de búsqueda secuencial.

0.- SALIR

```

//Martin Santos Tirado

//00369705
//06 de octubre de 2023
/*1.- LLENAR VECTOR .- Llenar vector con 15 números, los números
generados aleatoriamente, los números entre el rango de 100 al 200 (no
repetidos)
2.- LLENAR MATRIZ .- Llenar la matriz de 4x4 con con números generados
aleatoriamente, números entre el rango de 1 al 16 (no repetidos)
3.- IMPRIMIR VECTOR .- Imprime el vector que se envíe, donde la función
recibe como parámetro el vector,tamaño, nombre del vector.
4.- IMPRIMIR MATRIZ.- Imprime la matriz sin importar el tamaño de la
matriz recibiendo como parámetros la matriz,
la cantidad de renglones y columnas, así como nombre que se le dará a la
matriz
5.- ORDENAR VECTOR.- Usar función que ordene el vector por el método de
ordenación de la Burbuja mejorada.
6.- BUSCAR VALOR EN VECTOR.- Buscar un valor en el vector usando el
método de búsqueda secuencial.
0.- SALIR */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <stdbool.h>

void llenar_vector(int v1[]);
void imprvector(int v1[]);
void llenar_matriz4x4(int m[][4]);
void impr_matriz(int m[][4]);
bool sinrepetir(int v1[],int n);
void ordenar_vect(int v1[],int tam);
bool sinrepetir_m(int m[][4],int i,int j,int n);
void buscar_valor(int v1[],int tam);
void menu();

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

void menu()
{
    int v1[15];
    int op,i;
    int m[4][4];
    int tam;
    system("CLS");
    do{

```

```

    printf("\tMENU\n1-Llenar 15 numeros en el vector\n2-Llenar matriz del
1 al 16\n3-Imprimir vector\n4-Imprimir matriz\n5-Ordenar Vector\n6-Buscar
valor en vector\n0-Salir\n");
    scanf("%d",&op);
    switch (op)
    {
        case 1:
            llenar_vector(v1);
            break;
        case 2:
            llenar_matriz4x4(m);
            break;
        case 3:
            imprvector(v1);
            break;
        case 4:
            impr_matriz(m);
            break;
        case 5:
            ordenar_vect(v1,tam);
            break;
        case 6:
            buscar_valor(v1,tam);
            break;
        case 0:
            printf("Saliendo del programa\n");
            break;
    }

} while (op!=0);
}
void llenar_vector(int v1[])
{
    int i,num;
    system("CLS");
    srand(time(NULL));
    printf("Escogiste llenar vector aleatoriamente\n");
    for (i=0;i<15;i++)
    {
        do
        {
            num=100+rand()%(201-100);
        } while (sinrepetir(v1,num));
        v1[i]=num;
    }
    system("PAUSE");
}

```

```

}
bool sinrepetir(int v1[],int n)
{
    int i;
    for (i = 0; i <15; i++)
    {
        if (n == v1[i])
        {
            return true;
        }
    }
    return false;
}

void imprvector(int v1[])
{
    int i;
    system("CLS");
    for(i=0;i<15;i++)
    {
        printf("%2d:[%3d]\n",i+1,v1[i]); //i+1 es para que no se vea el
cero
    }
    system("PAUSE");
}

bool sinrepetir_m(int m[][4],int i,int j,int num)
{
    for (i=0;i<4;i++)
    {
        for (j=0;j<4;j++)
        {
            if(num == m[i][j])
            {
                return false;
            }
        }
    }
    return true;
}

void llenar_matriz4x4(int m[][4])
{
    int i,j,k;
    system("CLS");
    srand(time(NULL));
    int num;
    printf("Escogiste matriz de 4x4\n");
    for (i=0;i<4;i++)
    {
        for (j=0;j<4;j++)
        {

```

```

        do
        {
            num = 1+rand()%(17-1);
        } while (sinrepetir_m(m,4,4,num) == false);
        m[i][j] = num;

    }
    printf("\n");
}
system("PAUSE");
}

void impr_matriz(int m[][4])
{
    int i,j;
    system("CLS");
    printf("Escogiste imprimir matriz\n");
    for (i = 0; i < 4; i++)
    {
        printf("%2d: ",i);
        for (j = 0; j < 4; j++)
        {
            printf("[%3d]",m[i][j]);
        }
        printf("\n");
    }
    system("PAUSE");
}

void ordenar_vect(int v1[],int tam)
{
    system("CLS");
    printf("Escogiste ordenar vector\n");
    int i, j,temp;
    for(i = 0; i < 15; i++)
    {
        for(j = 0; j < 15; j++)
        {
            if (v1[j] > v1[j+1])
            {
                temp = v1[j+1];
                v1[j+1] = v1[j];
                v1[j] = temp;
            }
        }
    }
    for( i = 0 ; i < 15 ; i++)
    {

```

```

        printf("%2d:[%3d]\n",i+1,v1[i]); //i+1 es para que no se vea el
cero
    }
    system("PAUSE");
}

void buscar_valor(int v1[],int tam)
{
    system("CLS");
    int i,dato;
    char band = 'F';
    i = 0;
    printf("Que numero deseas buscar?\n");
    scanf("%d",&dato);
    while (band == 'F' && i < 15)
    {
        if (v1[i] == dato)
        {
            band = 'V';
        }
        i ++;
    }
    if (band == 'F')
    {
        printf("El numero a buscar no existe en el vector\n");
    }
    else if (band == 'V')
    {
        printf("El numero a sido encontrado en la posicion: %d\n",i);
    }
    system("PAUSE");
}

```

```

○      MENU
1-Llenar 15 numeros en el vector
2-Llenar matriz del 1 al 16
3-Imprimir vector
4-Imprimir matriz
5-Ordenar Vector
6-Buscar valor en vector
0-Salir

```

```

○ Escogiste llenar vector aleatoriamente
Press any key to continue . . .

```


○ Escogiste matriz de 4x4

Press any key to continue . . .

○ 1:[126]

2:[135]

3:[196]

4:[148]

5:[177]

6:[124]

7:[143]

8:[182]

9:[118]

10:[121]

11:[186]

12:[161]

13:[146]

14:[152]

15:[198]

Press any key to continue . . .

○ Escogiste imprimir matriz

0: [6][9][15][11]

1: [10][7][5][1]

2: [3][12][2][13]

3: [4][14][16][8]

Press any key to continue . . .

○ Escogiste ordenar vector

1:[118]

2:[121]

3:[124]

4:[126]

5:[135]

6:[143]

7:[146]

8:[148]

9:[152]

10:[161]

11:[177]

12:[182]

13:[186]

14:[196]

15:[198]

Press any key to continue . . .

Que numero deseas buscar?

161

El numero a sido encontrado en la posicion: 10

Press any key to continue . . .