

```

/*
Eliel Alfonso Ontiveros Ojeda_368746
29-Sep-2023 / 30-Sep-2023
Programa en el cual se le pide al usuario ingresar 10 numeros del 30 al 70 en el vector 1,
se generan 10 numeros aleatorios del 1 al 20 en el vector 2 y el vector 3 se rellena con los numeros del vector 1 y 2;
luego imprime el vector 3. Despues se seleccionan los primeros 4 numeros de los vectores para poder crear una matriz 4x4.
OOEA_Act8_1_932
*/

#include <stdio.h> // Incluye la biblioteca estándar de entrada y salida
#include <string.h> // Incluye la biblioteca de manejo de cadenas
#include <stdlib.h> // Incluye la biblioteca estándar
#include <time.h> // Incluye la biblioteca de tiempo

// Declaración de funciones
void manual(int vect1[10]); // Función para llenar manualmente el vector 1
void random(int vect2[10]); // Función para llenar aleatoriamente el vector 2
void vect1_vect2(int vect1[10], int vect2[10], int vect3[20]); // Función para llenar el vector 3 con los vectores 1 y 2
void imp_vectores(int vect1[10], int vect2[10], int vect3[20]); // Función para imprimir los vectores
void matriz_4x4(int matrix[30][30], int vect1[10], int vect2[10]); // Función para llenar una matriz 4x4 con los primeros 4 números de los
vectores
void imp_matriz(int matrix[30][30]); // Función para imprimir la matriz
int msgs(); // Función para imprimir el menú y obtener la opción del usuario
void menu(); // Función para manejar el menú

int main() // Función principal
{
    menu(); // Llama a la función del menú

    return 0; // Termina el programa
}

int msgs() // Función para imprimir el menú y obtener la opción del usuario
{
    int op; // Variable para almacenar la opción del usuario
    system("CLS"); // Limpia la pantalla
    printf("-----MENU-----\n"); // Imprime el menú
    printf("1 - Llenar Vector 1 (Manualmente)\n");
    printf("2 - Llenar Vector 2 (Aleatoriamente)\n");
    printf("3 - Llenar Vector 3 (Con Vector 1 y Vector 2)\n");
    printf("4 - Imprimir Verctores\n");
    printf("5 - Llenar Matriz 4 x 4\n");
    printf("6 - Imprimir Matriz\n");
    printf("0 - Salir\n");
    scanf("%d", &op); // Obtiene la opción del usuario

    return op; // Devuelve la opción del usuario
}

void menu() // Función para manejar el menú
{
    int op; // Variable para almacenar la opción del usuario
    int vect1[10], vect2[10], vect3[20], matrix[30][30]; // Declaración de los vectores y la matriz
    system("CLS"); // Limpia la pantalla

    do // Ciclo do-while para repetir el menú hasta que el usuario decida salir
    {
        op = msgs(); // Llama a la función msgs para obtener la opción del usuario

        switch (op) // Estructura switch para manejar las opciones del usuario
        {
            case 1:
                manual(vect1); // Llama a la función manual para llenar el vector 1
                break;

            case 2:
                random(vect2); // Llama a la función random para llenar el vector 2
                break;

            case 3:
                vect1_vect2(vect1, vect2, vect3); // Llama a la función vect1_vect2 para llenar el vector 3
                break;

            case 4:
                imp_vectores(vect1, vect2, vect3); // Llama a la función imp_vectores para imprimir los vectores
                break;

            case 5:
                matriz_4x4(matrix, vect1, vect2); // Llama a la función matriz_4x4 para llenar la matriz
                break;

            case 6:
                imp_matriz(matrix); // Llama a la función imp_matriz para imprimir la matriz
                break;

            default:

```

```

        break;
    }

} while (op != 0); // Repite el menú hasta que el usuario ingrese 0

printf("Saliendo del Programa"); // Imprime un mensaje de salida
}

void manual(int vect1[10]) // Función para llenar manualmente el vector 1
{
    printf("Llenar el vector 1 manualmente con numeros del 30 al 70:\n"); // Imprime las instrucciones
    for(int i = 1; i < 11; i++) // Ciclo for para recorrer el vector
    {
        printf("Numero %d\n", i); // Imprime el número actual
        scanf("%d", &vect1[i]); // Obtiene el número del usuario
    }

    system("PAUSE"); // Pausa el programa hasta que el usuario presione una tecla
}

void random(int vect2[10]) // Función para llenar aleatoriamente el vector 2
{
    srand(time(0)); // Inicializa la semilla del generador de números aleatorios

    for(int i = 0; i < 10; i++) // Ciclo for para recorrer el vector
    {
        vect2[i] = (rand() % 20) + 1; // Genera un número aleatorio entre 1 y 20 y lo guarda en el vector
    }

    system("PAUSE"); // Pausa el programa hasta que el usuario presione una tecla
}

void vect1_vect2(int vect1[10], int vect2[10], int vect3[20]) // Función para llenar el vector 3 con los vectores 1 y 2
{
    for(int i = 0; i < 10; i++) // Ciclo for para recorrer los vectores
    {
        vect3[i] = vect1[i]; // Copia el valor del vector 1 al vector 3
        vect3[i + 10] = vect2[i]; // Copia el valor del vector 2 al vector 3
    }

    system("PAUSE"); // Pausa el programa hasta que el usuario presione una tecla
}

void imp_vectores(int vect1[10], int vect2[10], int vect3[20]) // Función para imprimir los vectores
{
    printf("Vector 1:\n"); // Imprime el título del vector 1
    for(int i = 0; i < 10; i++) // Ciclo for para recorrer el vector 1
    {
        printf("%d ", vect1[i]); // Imprime el valor actual del vector 1
    }
    printf("\nVector 2:\n"); // Imprime el título del vector 2

    for(int i = 0; i < 10; i++) // Ciclo for para recorrer el vector 2
    {
        printf("%d ", vect2[i]); // Imprime el valor actual del vector 2
    }
    printf("\nVector 3:\n"); // Imprime el título del vector 3

    for(int i = 0; i < 20; i++) // Ciclo for para recorrer el vector 3
    {
        printf("%d ", vect3[i]); // Imprime el valor actual del vector 3
    }
    printf("\n"); // Imprime una línea en blanco

    system("PAUSE"); // Pausa el programa hasta que el usuario presione una tecla
}

void matriz_4x4(int matrix[30][30], int vect1[10], int vect2[10]) // Función para llenar una matriz 4x4 con los primeros 4 números de los vectores
{
    for(int i = 0; i < 4; i++) // Ciclo for para recorrer las filas de la matriz
    {
        for(int j = 0; j < 4; j++) // Ciclo for para recorrer las columnas de la matriz
        {
            if(i < 2) // Si la fila es menor a 2
            {
                matrix[i][j] = vect1[i * 4 + j]; // Llena la matriz con los valores del vector 1
            } else // Si la fila es mayor o igual a 2
            {
                matrix[i][j] = vect2[(i - 2) * 4 + j]; // Llena la matriz con los valores del vector 2
            }
        }
    }

    system("PAUSE"); // Pausa el programa hasta que el usuario presione una tecla
}

```

```
void imp_matriz(int matrix[30][30]) // Función para imprimir la matriz
{
    printf("Matriz:\n"); // Imprime el título de la matriz

    for(int i = 0; i < 4; i++) // Ciclo for para recorrer las filas de la matriz
    {
        for(int j = 0; j < 4; j++) // Ciclo for para recorrer las columnas de la matriz
        {
            printf("%d ", matrix[i][j]); // Imprime el valor actual de la matriz
        }
        printf("\n"); // Imprime una línea en blanco
    }

    system("PAUSE"); // Pausa el programa hasta que el usuario presione una tecla
}
```