#include <iostream>

#include <vector>

#include "estructuraM.h"

#include "cerradura.h"

#include "modulo5.cpp"

using namespace std;

int main() {

// Ejemplo de uso del módulo para crear una estructura M de tamaño variable con espacio en blanco en el centro

int tamano;

cout << "Ingrese el tamaño de la matriz: ";

cin >> tamano;

EstructuraM M = crearEstructuraM(tamano);

// Impresión de la estructura creada (solo como ejemplo)

cout << "Estructura M creada de tamaño " << M.filas << "x" << M.columnas << " con espacio en blanco en el centro:" << endl;

imprimirEstructuraM(M);

int numRotaciones;

cout << "\nIngrese el número de rotaciones que desea realizar: ";

cin >> numRotaciones;

rotarAntihorario(M, numRotaciones);

cout << "\nEstructura M rotada " << numRotaciones << " veces en sentido antihorario:" << endl;

imprimirEstructuraM(M);

// Liberar memoria asignada a la estructura M

liberarEstructuraM(M);

// Configurar una cerradura (Punto 3)

Cerradura cerradura = configureCerradura();

// Ejemplo de regla K

vector<int> regla = {4, 3, 1, -1, 1, 4, 2, 2, 1, 3};

// Validar la regla K sobre la cerradura (Punto 4)

if (validarRegla(cerradura, regla)) {

cout << "La regla se cumple para la cerradura configurada." << endl;

} else {

cout << "La regla no se cumple para la cerradura configurada." << endl;

}

// Liberar memoria asignada a las estructuras M de la cerradura

for (EstructuraM& estructura : cerradura.estructuras) {

liberarEstructuraM(estructura);

}

// Utilizar el módulo 5 para generar una configuración de cerradura que cumpla con la regla K

vector<int> reglaK = {4, 3, 1, 1, 2, 2, 1, -1, 3};

Cerradura configuracionValida = generarConfiguracionCerradura(reglaK);

// Aquí puedes imprimir o procesar la configuración válida obtenida

// ...

return 0;

}