**Solución**

Ejercicio 1

#include <iostream>

#include <stack>

using namespace std;

// Función para insertar un número en la pila

void insertarPila(stack<int>& pila) {

int num;

cout << "Ingrese un número entero: ";

cin >> num;

pila.push(num); // Se inserta el número en la pila

cout << "Número " << num << " insertado en la pila.\n";

}

// Función para eliminar el elemento superior de la pila

void borrarPila(stack<int>& pila) {

if (!pila.empty()) {

cout << "Número " << pila.top() << " eliminado de la pila.\n";

pila.pop(); // Se elimina el elemento en la cima de la pila

} else {

cout << "La pila está vacía, no hay elementos para eliminar.\n";

}

}

// Función para listar los elementos de la pila sin modificarla

void listarPila(stack<int> pila) {

if (pila.empty()) {

cout << "La pila está vacía.\n";

return;

}

cout << "Elementos en la pila (de arriba hacia abajo):\n";

while (!pila.empty()) {

cout << pila.top() << "\n"; // Se muestra el elemento en la cima

pila.pop(); // Se elimina temporalmente para mostrar el siguiente

}

}

int main() {

stack<int> pila; // Declaración de la pila de números enteros

int opcion;

do {

// Menú de opciones

cout << "\nMenú de gestión de pilas:\n";

cout << "1. Insertar en la pila\n";

cout << "2. Borrar de la pila\n";

cout << "3. Listar pila\n";

cout << "4. Salir\n";

cout << "Seleccione una opción: ";

cin >> opcion;

// Se ejecuta la opción elegida por el usuario

switch (opcion) {

case 1:

insertarPila(pila);

break;

case 2:

borrarPila(pila);

break;

case 3:

listarPila(pila);

break;

case 4:

cout << "Saliendo del programa...\n";

break;

default:

cout << "Opción no válida, intente de nuevo.\n";

}

} while (opcion != 4); // Se repite hasta que el usuario elija salir

return 0;

}

Explicación

Este programa en C++ permite gestionar una pila de números enteros mediante un menú interactivo con las siguientes opciones:

Insertar un número en la pila → Se usa push(num).

Eliminar el número superior de la pila → Se verifica si está vacía y se usa pop().

Listar los elementos de la pila → Se muestra el contenido desde arriba hacia abajo sin modificar la pila original.

Salir del programa.

El programa utiliza la estructura de datos pila (stack), que funciona con el principio LIFO (Último en entrar, primero en salir). Se implementa un do-while para mantener el menú activo hasta que el usuario decida salir.

**Funciones principales:**

push(num): Agrega un número.

pop(): Elimina el último número agregado.

top(): Muestra el número superior sin eliminarlo.

empty(): Verifica si la pila está vacía.

Ejercicio 2

#include <iostream>

#include <queue>

using namespace std;

// Función para mostrar el menú de opciones

void mostrarMenu() {

cout << "\nMENU:" << endl;

cout << "1. Insertar en cola" << endl;

cout << "2. Borrar en cola" << endl;

cout << "3. Listar cola" << endl;

cout << "4. Salir" << endl;

cout << "Seleccione una opcion: ";

}

int main() {

queue<int> cola; // Declaración de una cola de enteros

int opcion, numero;

do {

mostrarMenu(); // Mostrar el menú de opciones

cin >> opcion; // Leer la opción del usuario

switch (opcion) {

case 1:

// Insertar un número en la cola

cout << "Ingrese un numero: ";

cin >> numero;

cola.push(numero); // Agregar el número al final de la cola

cout << "Numero insertado en la cola." << endl;

break;

case 2:

// Borrar el primer elemento de la cola

if (!cola.empty()) {

cout << "Numero eliminado: " << cola.front() << endl; // Mostrar el número eliminado

cola.pop(); // Eliminar el primer elemento de la cola

} else {

cout << "La cola esta vacia, no hay elementos para borrar." << endl;

}

break;

case 3:

// Listar los elementos de la cola sin modificarlos

if (!cola.empty()) {

queue<int> temp = cola; // Crear una copia temporal de la cola

cout << "Elementos en la cola: ";

while (!temp.empty()) {

cout << temp.front() << " "; // Mostrar el primer elemento

temp.pop(); // Eliminarlo temporalmente para mostrar el siguiente

}

cout << endl;

} else {

cout << "La cola esta vacia." << endl;

}

break;

case 4:

cout << "Saliendo del programa..." << endl;

break;

default:

cout << "Opcion invalida. Intente de nuevo." << endl;

}

} while (opcion != 4); // Repetir hasta que el usuario elija salir

return 0;

}

explicación

Este programa en C++ implementa una cola de números enteros y permite gestionarla mediante un menú con cuatro opciones:

Insertar en la cola → Usa push(num) para agregar un número al final.

Eliminar de la cola → Si no está vacía, usa front() para mostrar el primer elemento y pop() para eliminarlo.

Listar la cola → Se usa una copia temporal para mostrar los elementos sin modificarlos.

Salir del programa.

Concepto clave:

Cola (queue) funciona con el principio FIFO (Primero en entrar, primero en salir)