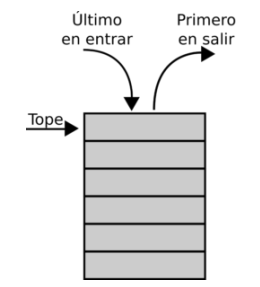
**Operaciones básicas con pilas**

**1. Introducción de Operaciones básicas con pilas**

Las pilas son estructuras de datos fundamentales en informática que siguen el principio de "último en entrar, primero en salir" (LIFO, por sus siglas en inglés). Las operaciones básicas con pilas son esenciales para manipular y gestionar estos elementos de manera eficiente.



***Figura****.1 Pila*

**2. Definición**

Una pila es una colección de elementos donde las operaciones de inserción y eliminación se realizan solo en un extremo llamado "cima" o "tope". Las operaciones básicas incluyen:

* Push: Agrega un elemento a la cima de la pila.
* Pop: Elimina el elemento en la cima de la pila.
* Peek o Top: Observa el elemento en la cima sin eliminarlo.
* isEmpty: Verifica si la pila está vacía.

**3. Importancia de las Operaciones básicas con pilas**

Las operaciones básicas con pilas son cruciales en muchas áreas de la informática, incluyendo algoritmos, compiladores, sistemas operativos y manejo de memoria. Proporcionan un mecanismo simple pero poderoso para gestionar datos de manera eficiente, permitiendo la reversión de operaciones y el control de flujo en muchas aplicaciones *Stein, C. (2009)*

**4. Uso de las Operaciones básicas con pilas**

Las pilas se utilizan en situaciones donde la reversión de operaciones o la gestión de tareas de manera inversa es necesaria. Algunos ejemplos comunes de uso incluyen:

* Evaluación de expresiones matemáticas.
* Implementación de algoritmos de búsqueda en profundidad.
* Manejo de llamadas a funciones en la ejecución de programas.
* Implementación de la funcionalidad "deshacer" en aplicaciones de software.

**5. Implementación de Operaciones básicas con pilas**

La implementación de operaciones básicas con pilas puede realizarse utilizando estructuras de datos como matrices o listas enlazadas. Es crucial mantener un seguimiento del tope de la pila y garantizar que las operaciones se realicen correctamente para evitar desbordamientos o subdesbordamientos *Tamassia, R. (2011).*

**#include <iostream>**

***// Definición de la clase NodoPila***

**class NodoPila {**

**public:**

**int dato;**

**NodoPila\* siguiente;**

**NodoPila(int dato) {**

**this->dato = dato;**

**this->siguiente = nullptr;**

**}**

**};**

***// Definición de la clase Pila***

**class Pila {**

**private:**

**NodoPila\* tope;**

**public:**

**Pila() {**

**tope = nullptr;**

**}**

**~Pila() {**

**while (!isEmpty()) {**

**pop();**

**}**

**}**

**bool isEmpty() {**

**return tope == nullptr;**

**}**

**void push(int dato) {**

**NodoPila\* nuevo\_nodo = new NodoPila(dato);**

**nuevo\_nodo->siguiente = tope;**

**tope = nuevo\_nodo;**

**}**

**int pop() {**

**if (isEmpty()) {**

**std::cerr << "Error: La pila está vacía\n";**

**return -1; *// Valor de retorno para indicar un error***

**}**

**int dato\_pop = tope->dato;**

**NodoPila\* temp = tope;**

**tope = tope->siguiente;**

**delete temp;**

**return dato\_pop;**

**}**

**int peek() {**

**if (isEmpty()) {**

**std::cerr << "Error: La pila está vacía\n";**

**return -1; *// Valor de retorno para indicar un error***

**}**

**return tope->dato;**

**}**

**};**

***// Función principal***

**int main() {**

**Pila pila;**

**std::cout << "La pila está vacía: " << std::boolalpha << pila.isEmpty() << "\n";**

**pila.push(10);**

**pila.push(20);**

**pila.push(30);**

**std::cout << "Elemento en la cima de la pila: " << pila.peek() << "\n";**

**std::cout << "Sacando elementos de la pila:\n";**

**while (!pila.isEmpty()) {**

**std::cout << pila.pop() << "\n";**

**}**

**std::cout << "La pila está vacía: " << std::boolalpha << pila.isEmpty() << "\n";**

**return 0;**

**}**

**6. Conclusiones**

Las operaciones básicas con pilas son fundamentales para la computación y ofrecen un mecanismo eficiente para gestionar datos en muchos contextos. Comprender y dominar estas operaciones es esencial para desarrollar algoritmos y aplicaciones eficientes y robustas.

**7. Referencias**

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms (3rd ed.)*. MIT Press.

Goodrich, M. T., & Tamassia, R. (2011). *Data Structures and Algorithms in Java (5th ed.)*. John Wiley & Sons.