



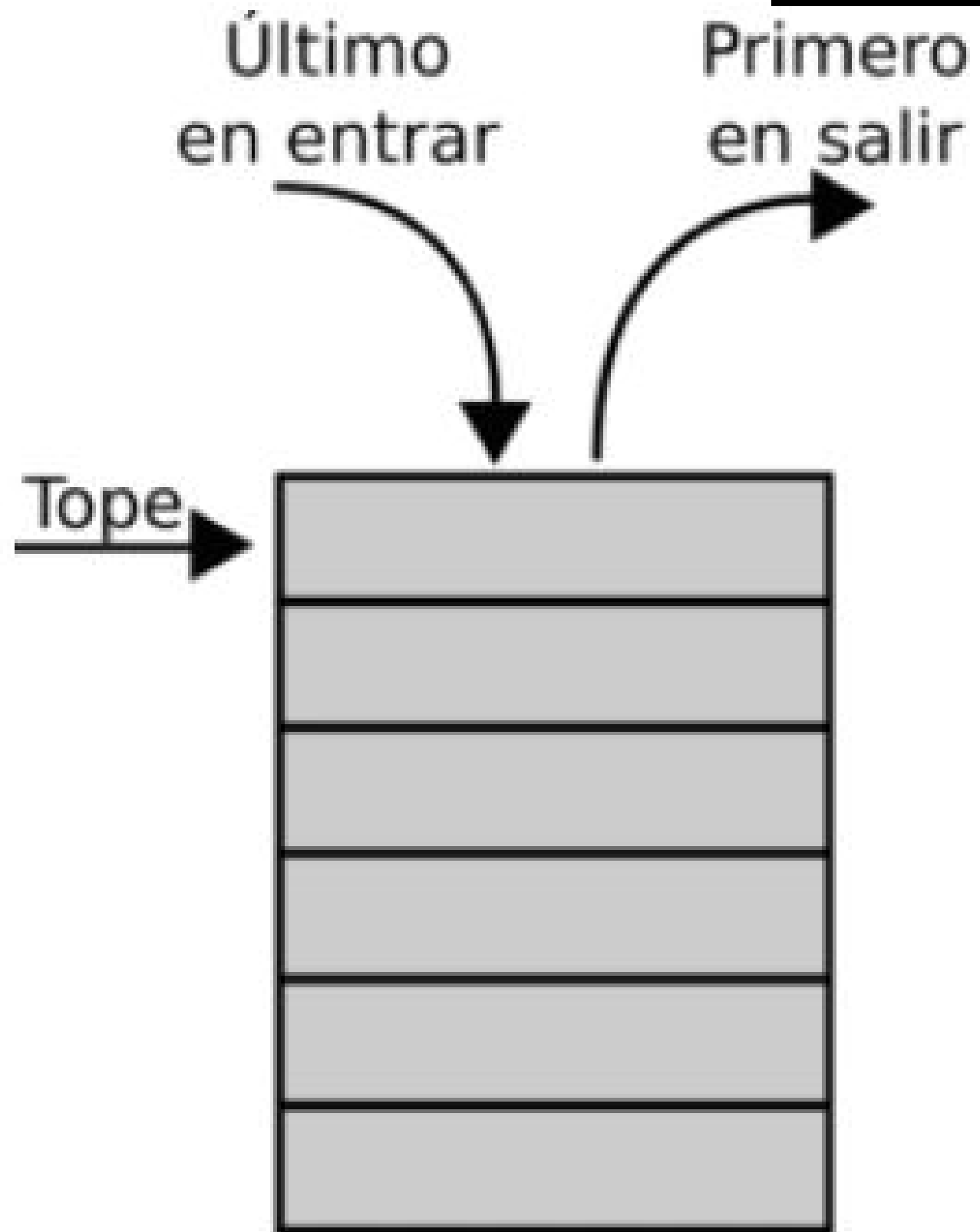
# Manejo de Pilas y Colas en Java

**Materia: Estructura de Datos**

# Que son?

Una pila (stack) es una colección ordenada de elementos a los cuales sólo se puede acceder por un único lugar o extremo de la pila. Los elementos se añaden o se quitan (borran) de la pila sólo por su parte superior (cima). Este es el caso de una pila de platos, una pila de libros, etc.

Cuando se dice que la pila está ordenada, lo que se quiere decir es que hay un elemento al que se puede acceder primero (el que está encima de la pila), otro elemento al que se puede acceder en segundo lugar (justo el elemento que está debajo de la cima), un tercero, etc. No se requiere que las entradas se puedan comparar utilizando el operador “menor que” ( $<$ ) y pueden ser de cualquier tipo.



# Pilas en Java

Las entradas de la pila deben ser eliminadas en el orden inverso al que se situaron en la misma. Por ejemplo, se puede crear una pila de libros, situando primero un diccionario, encima de él una enciclopedia y encima de ambos una novela, de modo que la pila tendrá la novela en la parte superior.

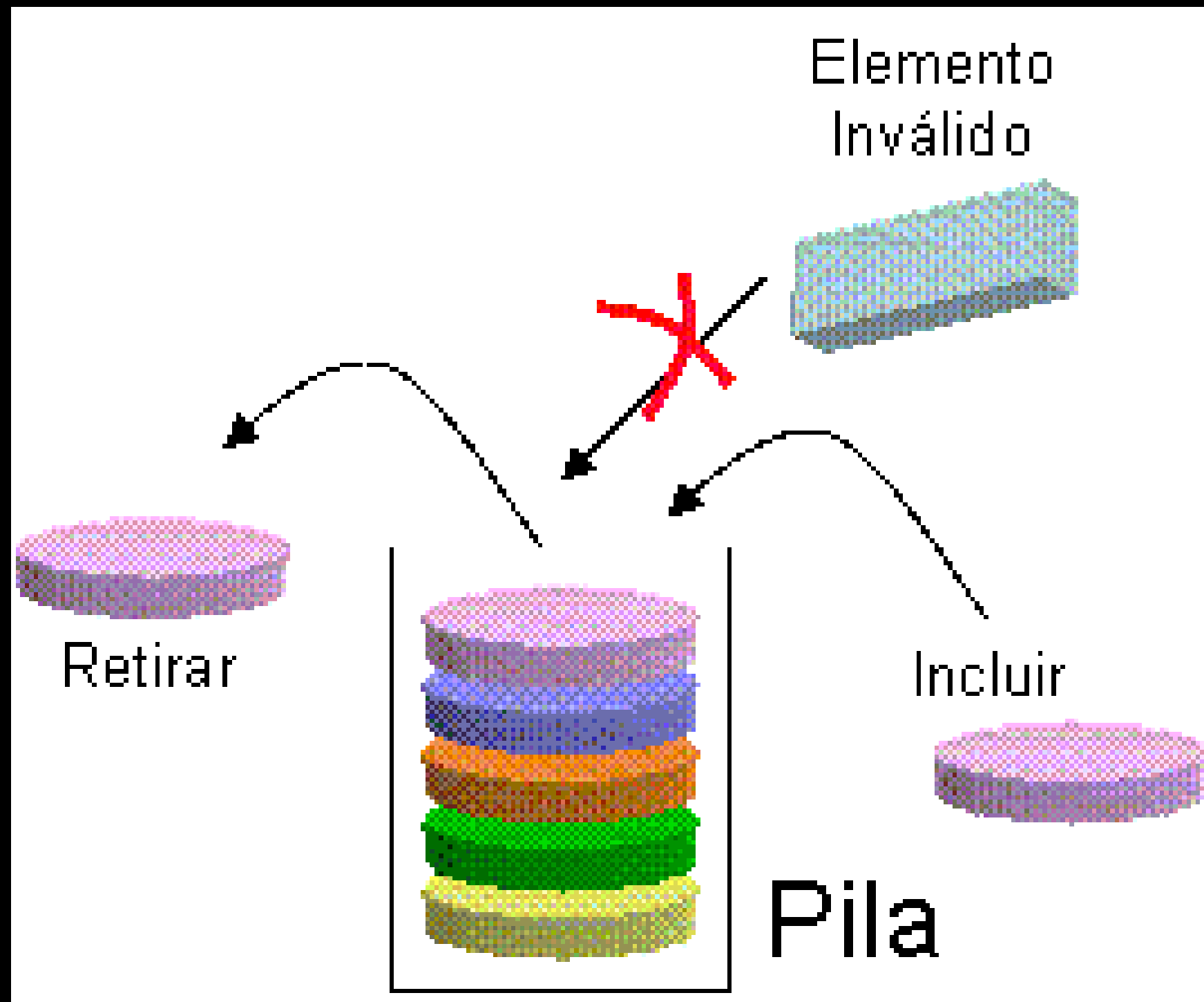
Cuando se quitan los libros de la pila, primero debe quitarse la novela, luego la enciclopedia y por último el diccionario.



# Operaciones en Pilas

Las operaciones fundamentales en **pilas** incluyen *push* (para agregar elementos), *pop* (para eliminar el elemento superior) y *peek* (para obtener el elemento superior sin eliminarlo).

Estas operaciones son clave en el manejo eficiente de datos en Java.



# Operaciones con Colas y Pilas

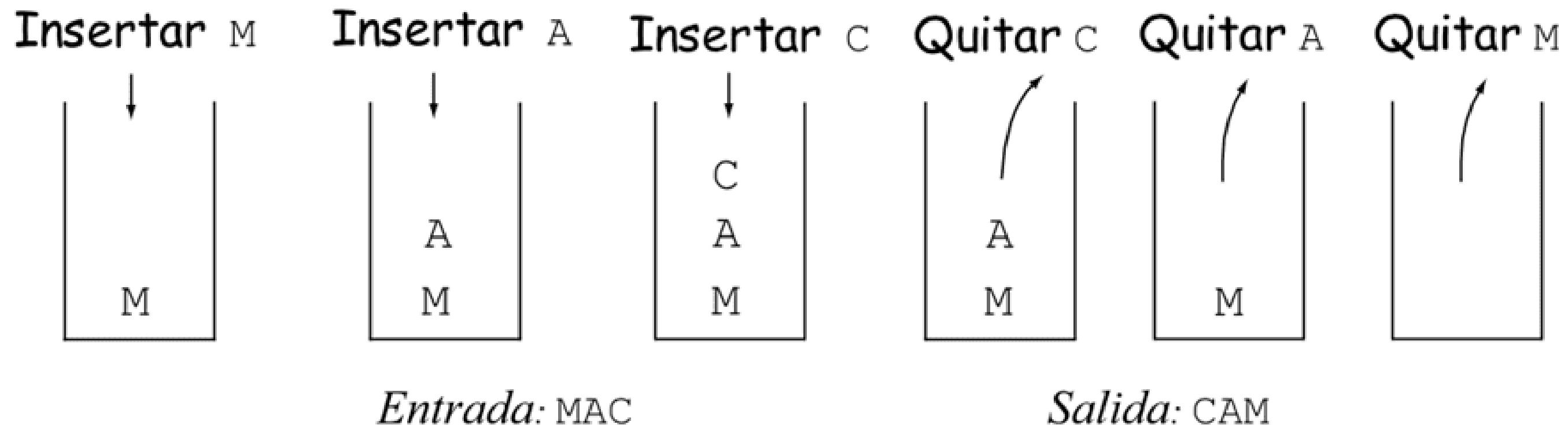
Debido a su propiedad específica último en entrar, primero en salir se conoce a las pilas como estructuras de datos LIFO (last-in, first-out)

Las operaciones usuales en la pila son

Insertar y Quitar. La operación Insertar (push) añade un elemento en la cima de la pila, y la operación Quitar (pop) elimina o saca un elemento de la pila.

La siguiente figura muestra una secuencia de operaciones Insertar y Quitar.

El último elemento añadido a la pila es el primero que se quita de ella.



# Operaciones en Colas



Una pila puede estar vacía (no tiene elementos) o llena (en la representación con un array —arreglo—, si se ha llegado al último elemento). Si un programa intenta sacar un elemento de una pila vacía, se producirá un error, una excepción, debido a que esa operación es imposible; esta situación se denomina desbordamiento negativo (underflow).

Por el contrario, si un programa intenta poner un elemento en una pila llena, se produce un error, una excepción, de desbordamiento (overflow) o rebosamiento. Para evitar estas situaciones se diseñan métodos que comprueban si la pila está llena o vacía. Los algoritmos de introducir, “insertar” (push) y “quitar”, sacar, (pop) datos de la pila son:

Los algoritmos de introducir, “insertar” (*push*) y “quitar”, sacar, (*pop*) datos de la pila son:

#### Insertar (*push*)

1. Verificar si la pila no está llena.
2. Incrementar en 1 el puntero índice de la pila.
3. Almacenar elemento en la posición del puntero de la pila.

#### Quitar (*pop*)

1. Verificar si la pila no está vacía.
2. Leer el elemento de la posición del puntero de la pila.
3. Decrementar en 1 el puntero de la pila.

# Conclusión

El **manejo de pilas y colas** en Java es esencial para la organización eficiente de datos en aplicaciones. Comprender el funcionamiento de estas estructuras de datos es crucial para el desarrollo de software robusto y eficiente.