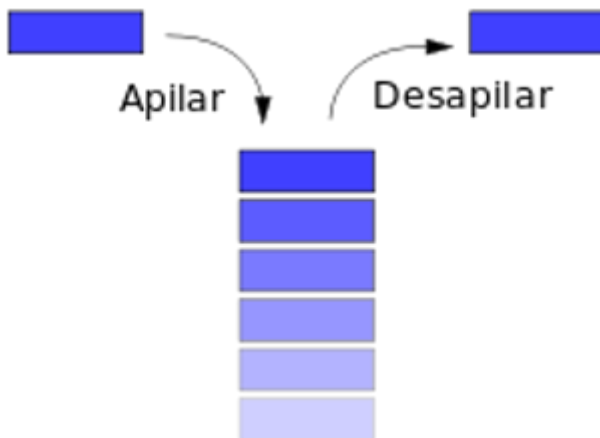


Pilas y Colas

Las pilas (stacks) y las colas (queues) son tipos abstractos de datos que nos permiten acceder a la información contenida en ellas de manera controlada, ya que solo se puede acceder a ella desde uno de los extremos (inicio o final). Tanto una pila como una cola pueden ser implementadas en una estructura de datos física como una lista enlazada o un vector, lo importante en cada caso es que las restricciones de acceso, inserción y borrado sean establecidas correctamente.

Pila (stack)



Las pilas son estructuras de datos lineales donde la restricción nos indica que la **inserción (apilar o push)** de un nuevo elemento solo debe realizarse desde la *cima* (o inicio) de la pila, y la **extracción o borrado (desapilar o pop)** de un elemento debe también realizarse desde la *cima* de la pila.

Este comportamiento hace que la pila sea una estructura tipo **LIFO** (*last in first out*, el último que entra es el primero que sale). En ciencias de la computación las pilas suelen usarse para:

- Evaluación de expresiones en notación postfija (notación polaca inversa).
- Reconocedores sintácticos de lenguajes independientes del contexto
- Implementación de algoritmos recursivos.
- Etc.

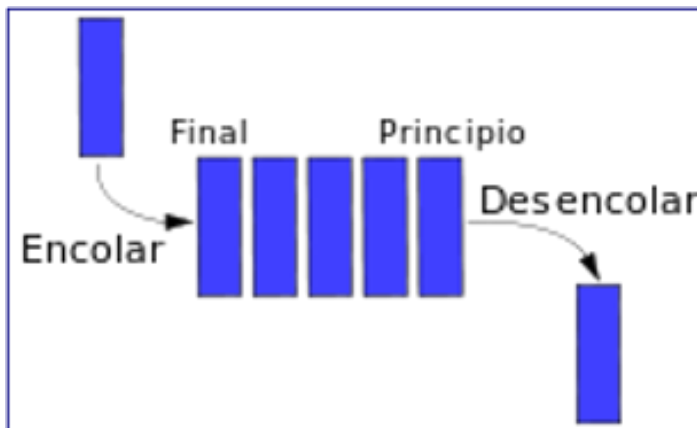
Operaciones

Las operaciones principales de una pila son **apilar (push)** y **desapilar (pop)**. Usualmente se suelen agregar las funciones **cima (top)**, **tamaño (size)** y **vacio (is_empty)** como una manera de darle un mayor soporte al programador al momento de manejar la pila.

- **apilar (push)**: inserta un elemento en la cima de la pila.
- **desapilar (pop)**: retorna el elemento en la cima de la pila y lo elimina de la estructura.
- **cima (top)**: retorna el elemento en la cima de la pila sin eliminarlo.
- **tamaño (size)**: retorna la cantidad de elementos de la pila.
- **vacio (is_empty)**: retorna verdadero si la pila no tiene elementos.

Además cuando una pila es implementada sobre un vector o arreglo se suele incluir una función que indique el tamaño máximo que puede tener la pila (**max_size**).

Cola (queue)



Las colas son estructuras de datos lineales donde la restricción nos indica que la **inserción (encolar o push)** de un nuevo elemento debe realizarse desde el final de la cola, y la **extracción o borrado (desencolar o pop)** de un elemento debe realizarse desde el inicio de la cola, es decir, ambas operaciones se realizan desde extremos opuestos de la estructura.

Este comportamiento hace que la cola sea una estructura tipo **FIFO** (*first in first out*, el primero que entra es el primero que sale). Las colas se utilizan en sistemas informáticos, transportes y operaciones de investigación (entre otros), donde los objetos, personas o eventos son tomados como datos que se almacenan y se guardan mediante colas para su posterior procesamiento.

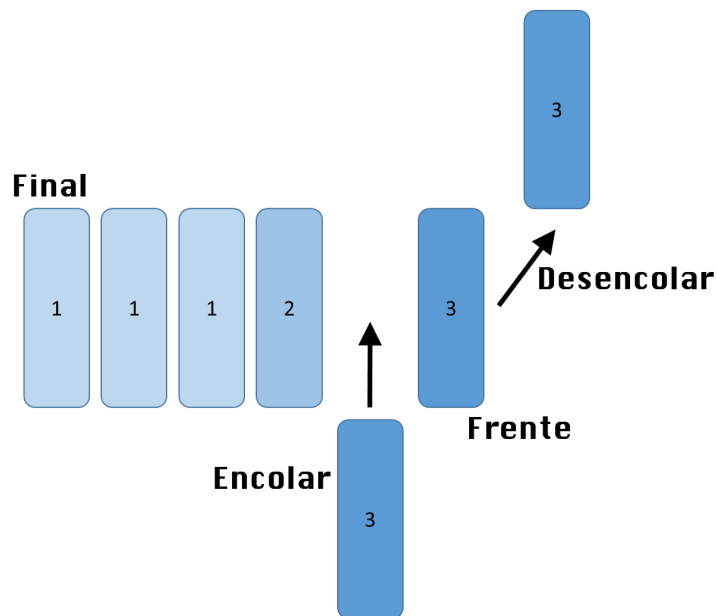
Operaciones

Las operaciones principales de una cola son **encolar (push)** y **desencolar (pop)**. Usualmente se suelen agregar las funciones **frente (front)**, **tamaño (size)** y **vacio (is_empty)** como una manera de darle un mayor soporte al programador al momento de manejar la cola.

- **encolar (push)**: inserta un elemento al final de la cola.
- **desencolar (pop)**: retorna el elemento del inicio de la cola y lo elimina de la estructura.
- **frente (front)**: retorna el elemento del inicio de la cola sin eliminarlo.
- **tamaño (size)**: retorna la cantidad de elementos de la cola.
- **vacio (is_empty)**: retorna verdadero si la cola no tiene elementos.

Además cuando una cola es implementada sobre un vector o arreglo se suele incluir una función que indique el tamaño máximo que puede tener la pila (**max_size**).

Cola de prioridad (priority queue)



Una cola de prioridades es un tipo de dato abstracto similar a una cola en la que los elementos tienen adicionalmente, una prioridad asignada. En una cola de prioridades un elemento con mayor prioridad será desencolado antes que un elemento de menor prioridad. Si dos elementos tienen la misma prioridad, se desencolarán siguiendo el orden de cola.

Operaciones

Una cola de prioridad ha de soportar al menos las siguientes dos operaciones:

- **encolar (con prioridad):** se añade un elemento a la cola, con su correspondiente prioridad.
- **desencolar (con prioridad):** se retorna y elimina el elemento con mayor prioridad más antiguo que no haya sido desencolado de la cola.

Además suele implementarse una función frente (que habitualmente aquí se denomina encontrar-máximo o encontrar-mínimo), y que habitualmente se ejecuta en tiempo $O(1)$. Esta operación y su rendimiento en tiempo es crucial en ciertas aplicaciones de las colas de prioridades.

Ciertas implementaciones avanzadas pueden incluir operaciones más complejas para la inspección de los elementos de mayor o menor prioridad, borrar la cola o ciertos subconjuntos de la cola, realizar inserciones en masa, la fusión de dos colas en una, aumentar la prioridad de los elementos, etc.