



Estructuras de Datos

IIS262

Profesor: Patricio Galeas

CAPÍTULO 4 : Pilas y Colas



Introducción

- Revisaremos 3 estructuras de datos:
 - Pilas
 - Colas
 - Colas de Prioridad
- Como estas difieren de los arreglos
- Algunas aplicaciones interesantes (parsing de expresiones aritméticas)
- Como vimos anteriormente, los arreglos son estructuras muy usadas en bases de datos (datos reales). Facilitando el acceso a los datos: inserción, búsqueda y eliminación.
- Las estructuras que veremos ahora
 - son más usadas como herramientas de programación. Y son creadas y usadas durante la ejecución y luego eliminadas.
 - Tienen un acceso más restringido (acceso de sólo un elemento a la vez)
 - Los estructuras para manejar pilas pueden ser arreglos o listas enlazadas
 - Los estructuras para colas de prioridad pueden ser arreglos y arboles (montículos)

Pilas



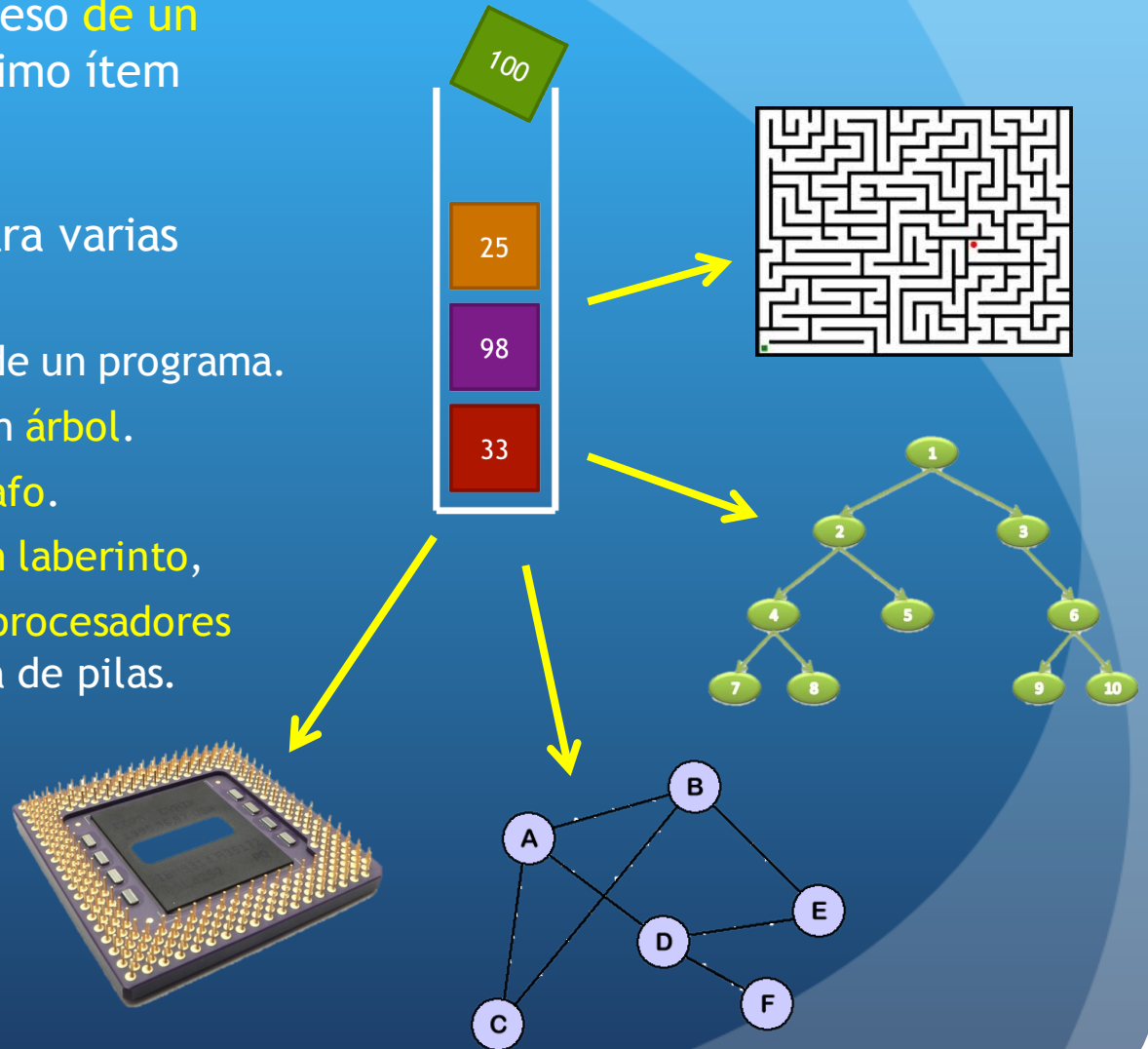
¿Que son las Pilas?

Pilas



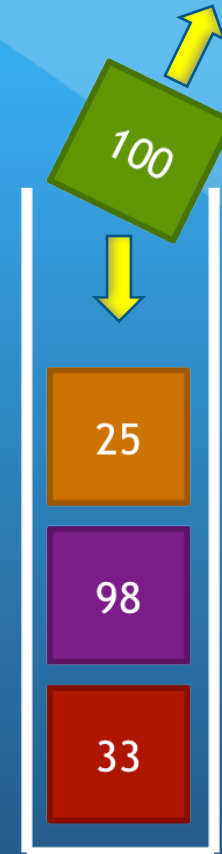
Introducción

- Las pilas permiten el acceso **de un elemento a la vez** (el último ítem insertado).
- Esta estructura es útil para varias situaciones:
 - **Balanceo de paréntesis** de un programa.
 - **Recorrido de nodos** en un árbol.
 - **Buscar vértices** en un grafo.
 - Encontrar la **salida en un laberinto**,
 - La mayoría de los **microprocesadores** utilizan una arquitectura de pilas.

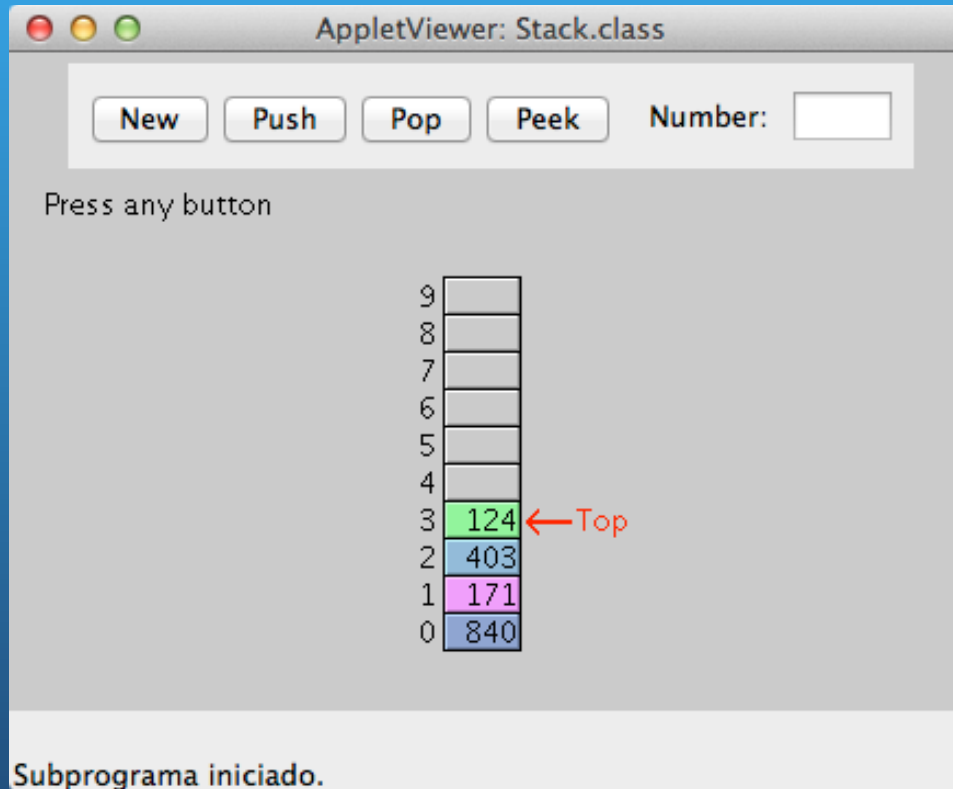


¿Que son las Pilas?

- El **procesamiento** de la pila es **de arriba hacia abajo** (top down)
- Una vez que el **primer elemento** de la pila es **procesado/eliminado** de ella, **se puede acceder al siguiente elemento**, que pasa a tomar la primera posición.
- Esta técnica **funciona bien, cuando hay tiempo para procesar todos los elementos** (riesgo: no alcanzar nunca el último elemento).
- Cuando se **agrega** un elemento a la pila se dice que se **“apila”** (push) y cuando **eliminamos** un elemento, lo **“desapilamos”** (pop). Para **leer** el primer elemento (sin eliminarlo) aplicamos el comando **“cima”** (peek).
- La Pilas es un mecanismo de almacenamiento del tipo **“Last In First Out”** (LIFO)



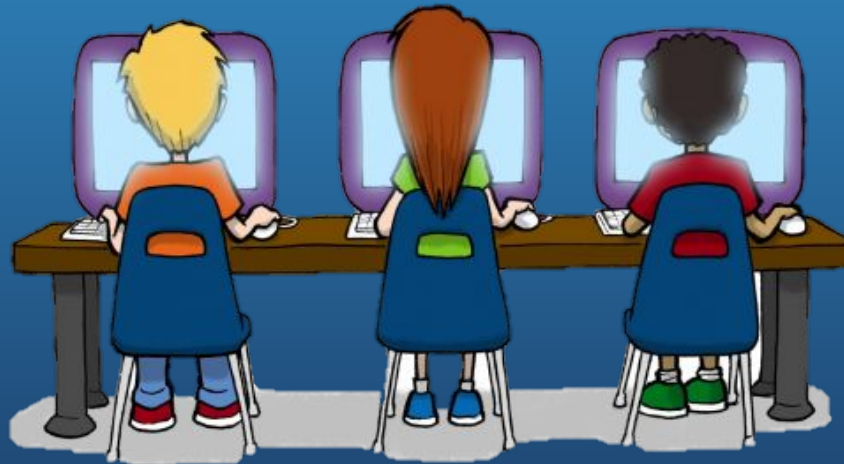
Funcionamiento de una Pila



applet

Implementación de una Pila en Java

Stack.java

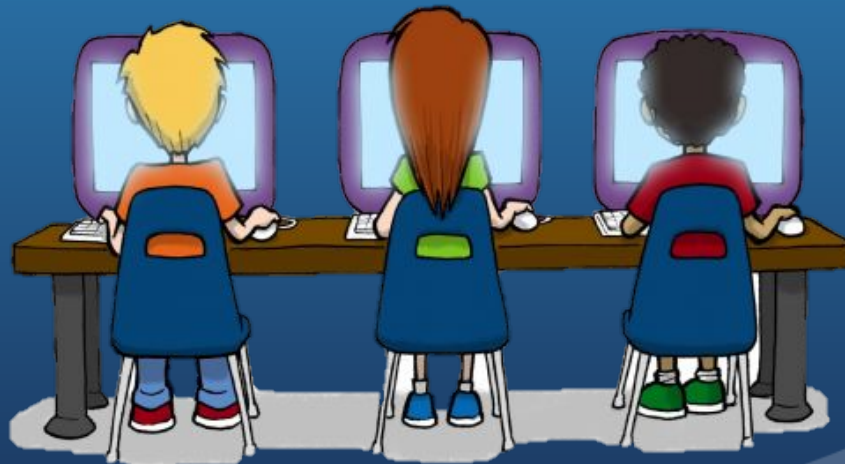


Algunos Ejemplos de Pilas



Invertir palabras
reverse.java

Delimitador de paréntesis
brackets.java



Eficiencia de Pilas

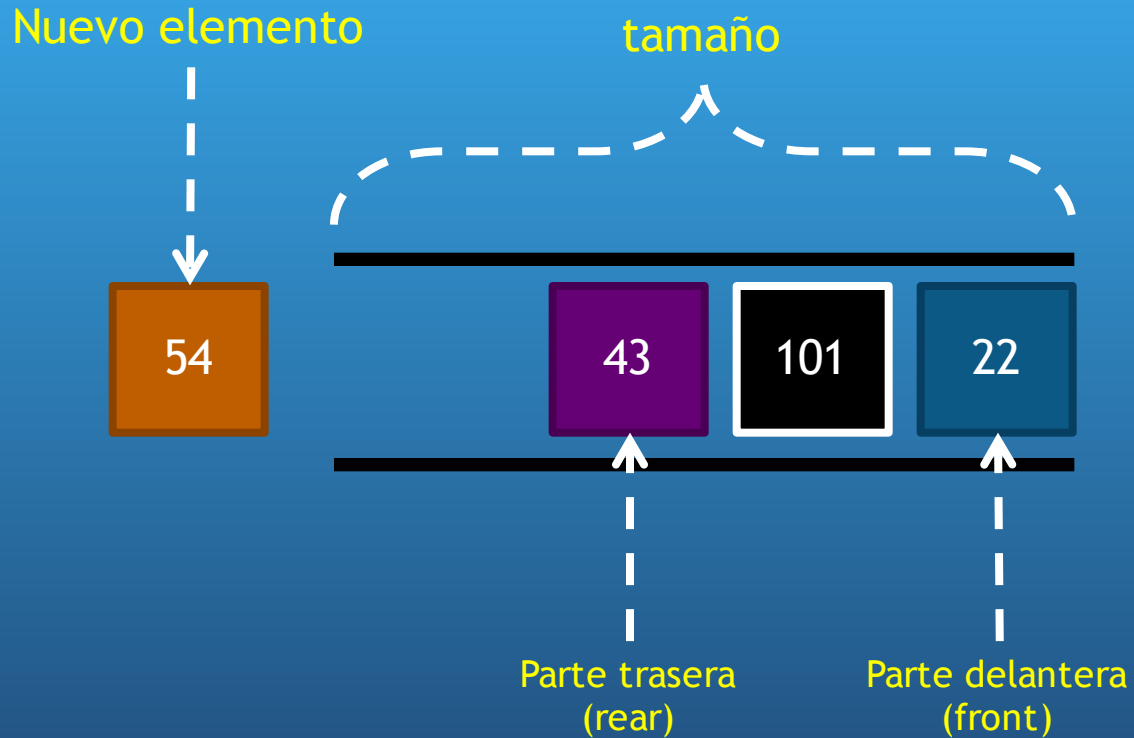
- Los ítems en una pila pueden ser apilados y desapilados en un **tiempo constante** : $O(1)$
- Es decir, el **tiempo no depende del número de ítems** que hay en la pila.
- No son necesarias **ni comparaciones ni movimientos** de ítems.
- Las Pilas son **muy rápidas!**



Colas



¿Que es una Cola?



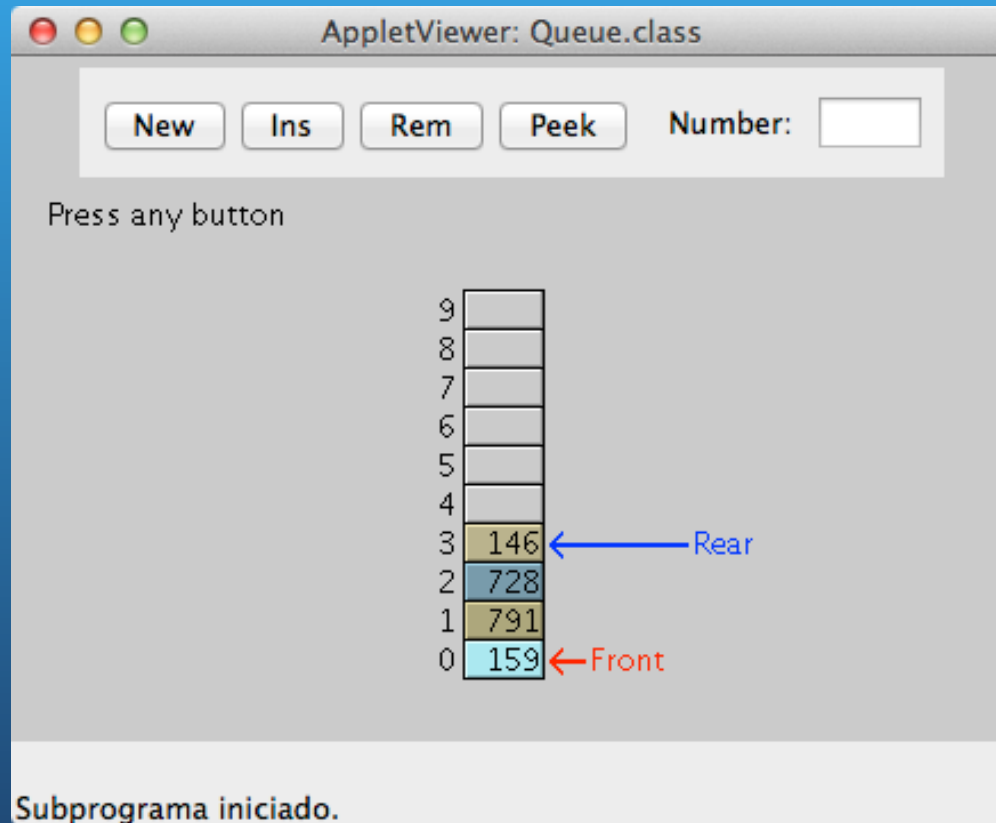
¿Que es una Cola?

- Las colas son parecidas a las pilas, con la diferencia de que es **FIFO (First In First Out)** : primer elemento en entrar es el primer elemento en salir
- Es el mismo concepto de cola en la vida real. **Por ejemplo**: para comprar una entrada al cine.
- Utilizadas (al igual que las pilas) como **herramientas de programación**.
- Utilizadas para modelar **situaciones reales**



- Banco
- Aviones esperando despegar
- Información esperando a ser transmitida a través de internet
- Sistemas Operativos (impresión)
- etc.

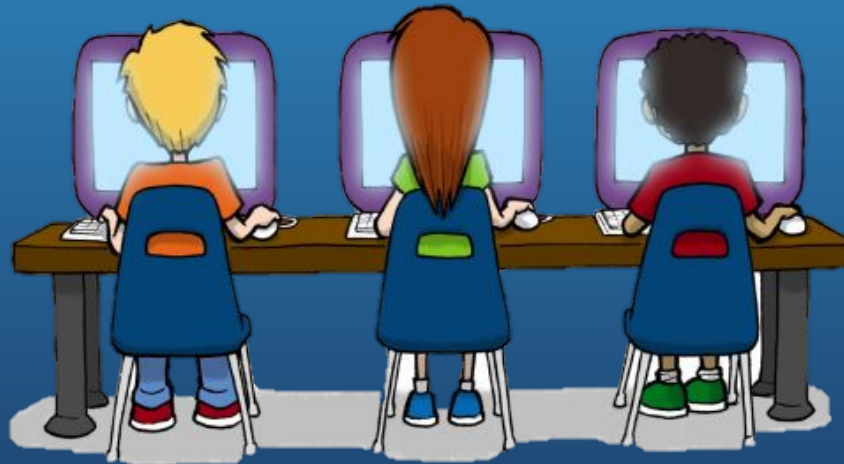
Funcionamiento de una Cola



applet

Implementación de una Cola en Java

queue.java

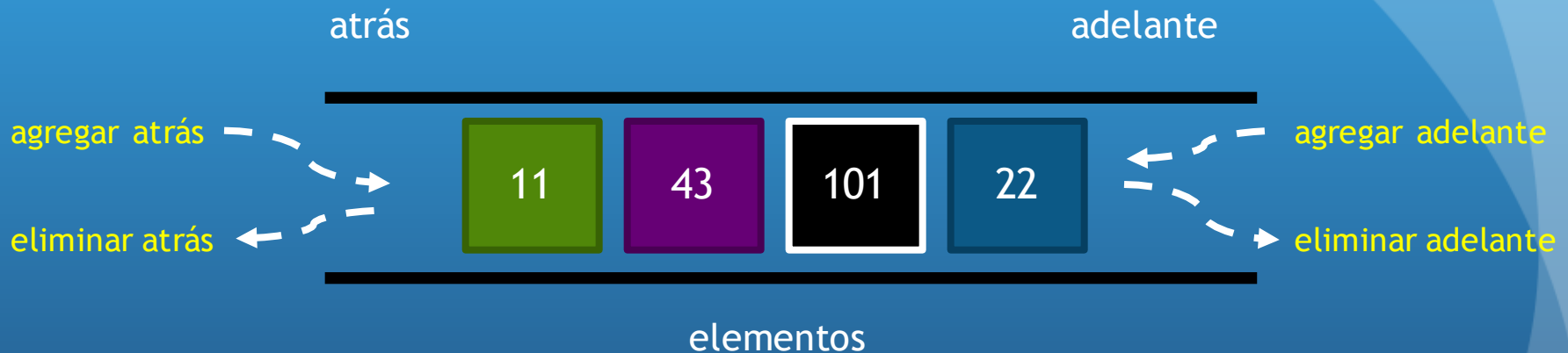


Eficiencia de Colas



- Los ítems en una cola pueden ser insertados y eliminados en un **tiempo constante** : $O(1)$
- Es decir, el **tiempo no depende del número de ítems** que hay en la cola.
- No son necesarias **ni comparaciones ni movimientos** de ítems.
- Las **colas** también son **muy rápidas!**

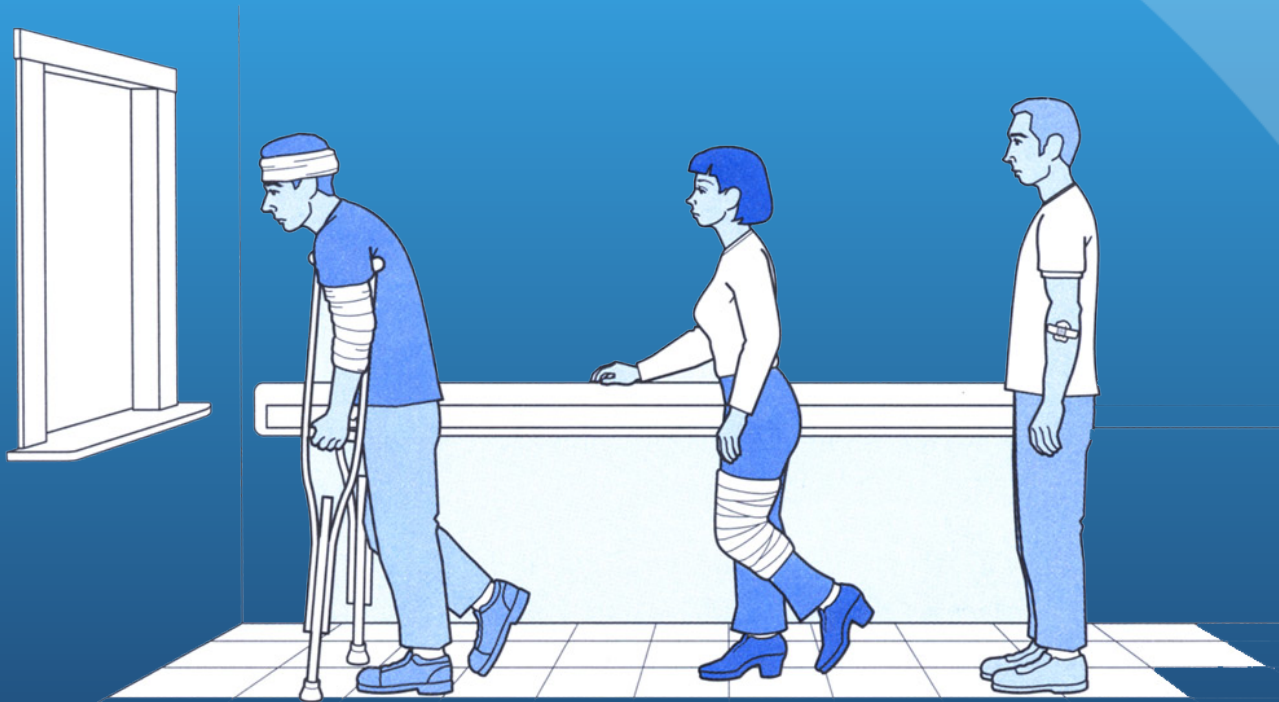
Cola Doblemente Terminada (Deque)



Si se restringe a

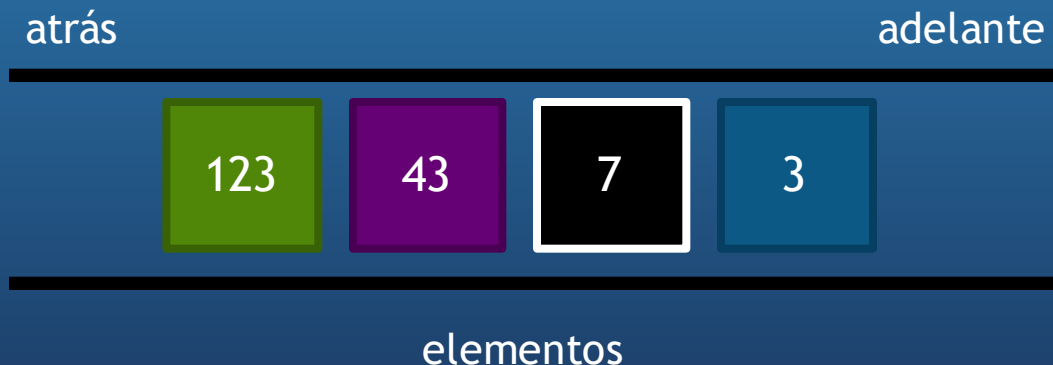
- agregar atrás, eliminar atrás => pila
- agregar adelante, eliminar adelante => pila
- agregar adelante, eliminar atrás => cola
- agregar atrás, eliminar adelante => cola

Colas de Prioridad

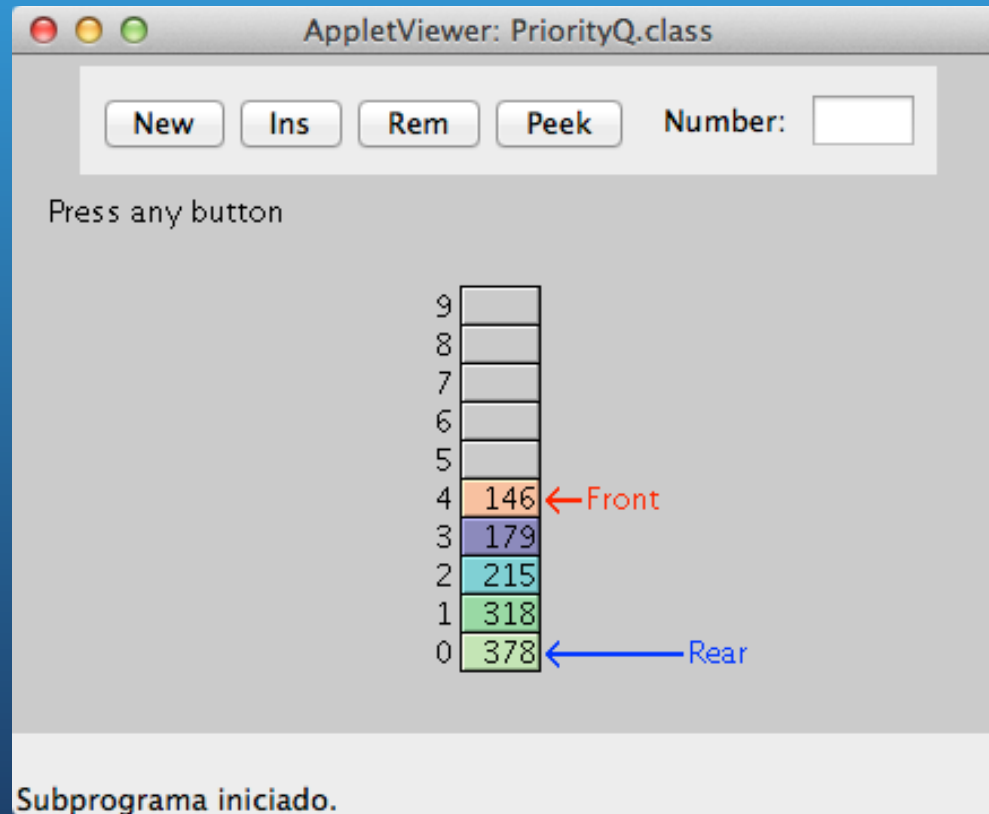


¿ Que es una Cola de Prioridad ?

- Es una estructura mas especializada que la cola simple y la pila.
- Al igual que la cola tiene un **front** y **rear**.
- Se diferencia de la cola simple, ya que **sus elementos están ordenados**.
- Se pueden utilizar en **varias situaciones**:
Ej. Priorizar procesos en un sistema operativo
- Las implementaremos como **arreglos**, pero también pueden ser implementadas con estructuras **Heap** (Montículos)



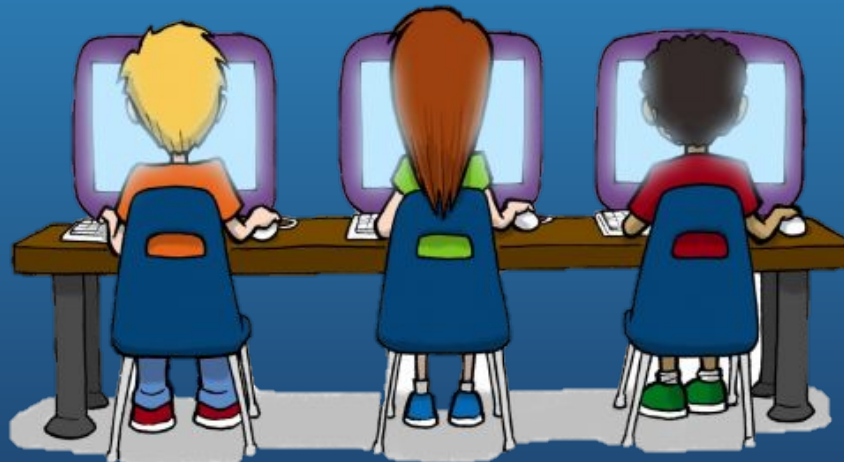
Funcionamiento Cola de Prioridad



applet

Implementación Cola de Prioridad en Java

PriorityQ.java



Eficiencia

Colas de Prioridad



- Los ítems en una cola de prioridad pueden **eliminados** en un **tiempo constante** : $O(1)$
- Los ítems en una cola de prioridad pueden **insertados** en un **tiempo**: $O(N)$
- La inserción se puede mejorar utilizando otra estructura (heap)

Resumen

- Pilas y colas son estructuras normalmente usadas para simplificar ciertas operaciones computacionales.
- En estas estructuras sólo un elemento puede ser accedido.
- **La pila**
 - permite el acceso al último elemento insertado.
 - Operaciones: push & pop, ambas se aplican a la cima (top)
- **La Cola**
 - Permite acceso al primer elemento insertado.
 - Operaciones: insert (atrás) & remove (adelante)
 - Puede ser implementada como una cola circular
- **La Cola de Prioridad**
 - Permite el acceso al menor (a veces mayor) elemento.
 - Operaciones: insertar (en forma ordenada), eliminar el elemento menor.
- Estas estructuras pueden ser implementadas con arreglos u otros mecanismos (listas enlazadas)



Experimentos

- Usando el applet Queue, remueva e inserte items en forma alternada, reusando el item eliminado. Se aprecia que un grupo de 4 elemento escala hasta la cima de la cola y reaparecen por abajo.
- Usando el applet PriorityQ, imagínese las flechas front & rear cuando la cola esta llena y cuando esta vacía. ¿Se puede implementar aquí algo parecido a una cola circular?
- Escriba un método para la clase Queue (queue.java), que despliegue el contenido de la cola. Notar que no se trata de simplemente imprimir el contenido del arreglo, sino que mostrar desde el primer al último elemento.



Experimentos

- Crear una clase para Dequeue, que incluya los métodos `insertLeft()`, `insertRight()`, `removeLeft()`, `removeRight()`, `isEmpty()`, and `isFull()`. La clase también debe soportar el desplazamiento circular, como lo hace el ejemplo de cola estándar.
- La cola de prioridad mostrada en clase implemente métodos para una rápida extracción de los elementos con alta prioridad pero una lenta inserción de nuevos ítems.

Escriba una nueva clase `PriorityQ` que tenga una inserción de orden casi $O(1)$ pero lenta eliminación de los ítems con alta prioridad. Incluya además un método para desplegar los elementos de la cola.

