Estructura de Datos y Algoritmos

Colas

Semana 5

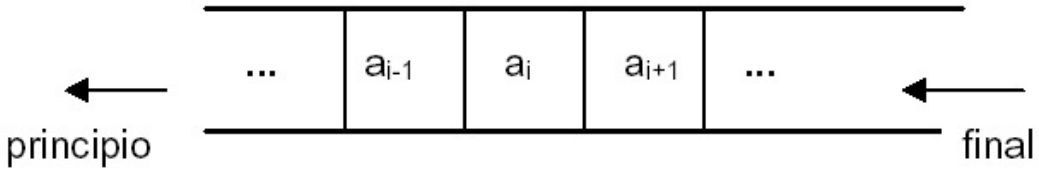
Agenda

- Definición de Cola (TDA Cola)
- Tipos de Cola
- Implementación de una cola

Cola

Una cola es una estructura de datos, caracterizada por ser una secuencia de elementos en la que la operación de inserción (push/poner) se realiza por un extremo y la operación de extracción (pop/sacar)se realiza por el otro.

También se le llama estructura **FIFO** (del inglés *First In First Out*), debido a que el primer elemento en entrar será también el primero en salir.

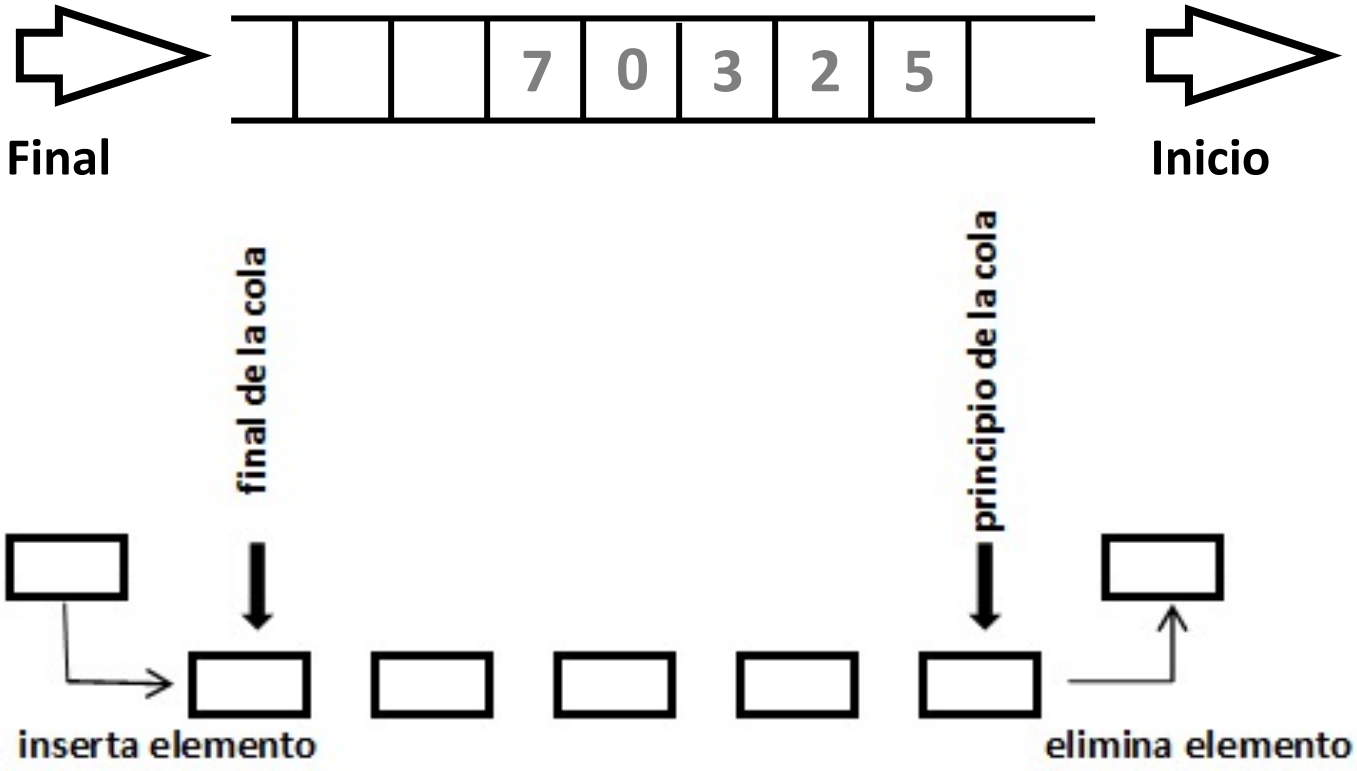


Cola

Ejemplo de colas:

- Cola para comprar las entradas del cine.
- Cola de Atención en el banco
- Cola para votar en las elecciones
- Cola de impresión.
- Cola de procesos en la CPU
- Cola en el cifrado y descifrado de mensajes

Cola



Los elementos se eliminan por el inicio y se insertan por el final.

Cola

Operaciones:

- Crear (cola).
- Está vacía (*isEmpty*): Determina si la cola está vacía
- Incluir (*enqueue*): Inserta un elemento al final de la cola.
- Eliminar (*dequeue*): Elimina un elemento del inicio de la cola.
- Acceso (*first*): Accede o examina al elemento que se encuentra en el inicio de la cola.

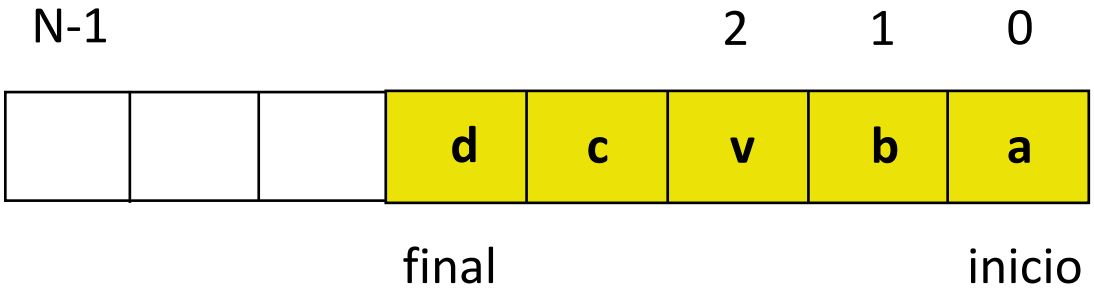
Cola

Formas de implementación:

- Arreglos
- Listas enlazadas

Cola

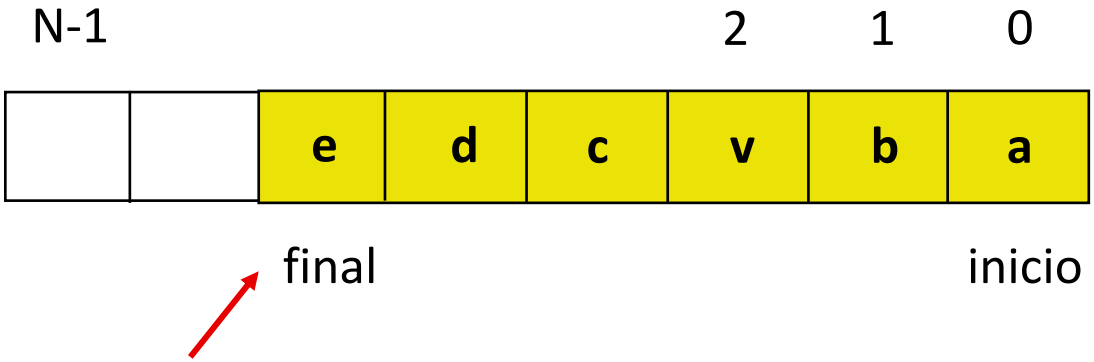
Usando arreglos:



Cola

Usando arreglos:

*Al adicionar un valor 'e', incrementamos final en 1 e insertamos el valor en la nueva posición **final***



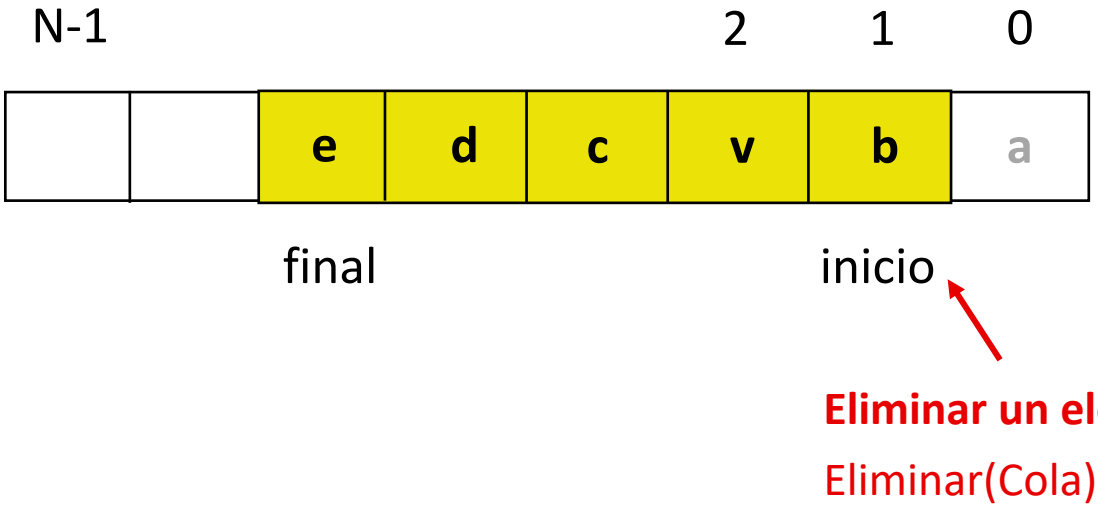
Incluir un elemento

Incluir('e', Cola)

Cola

Usando arreglos:

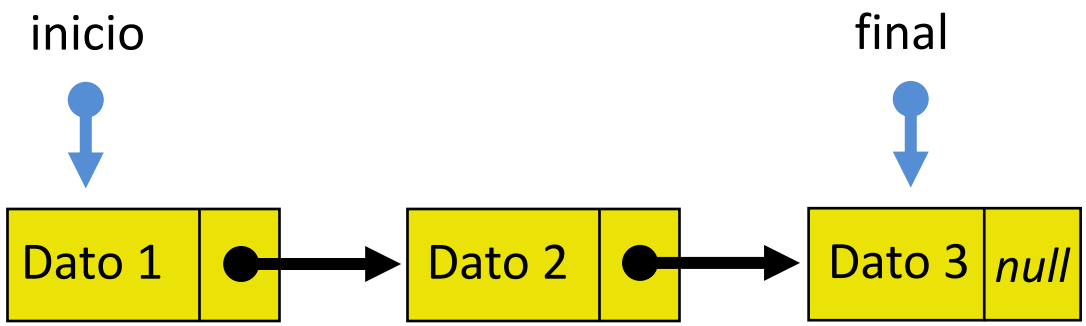
*Al eliminar, retornamos el valor en **inicio** e incrementamos **inicio** en 1*



Cola

Usando listas enlazadas:

- La adición es por el final de la lista y la eliminación es por el inicio de la lista

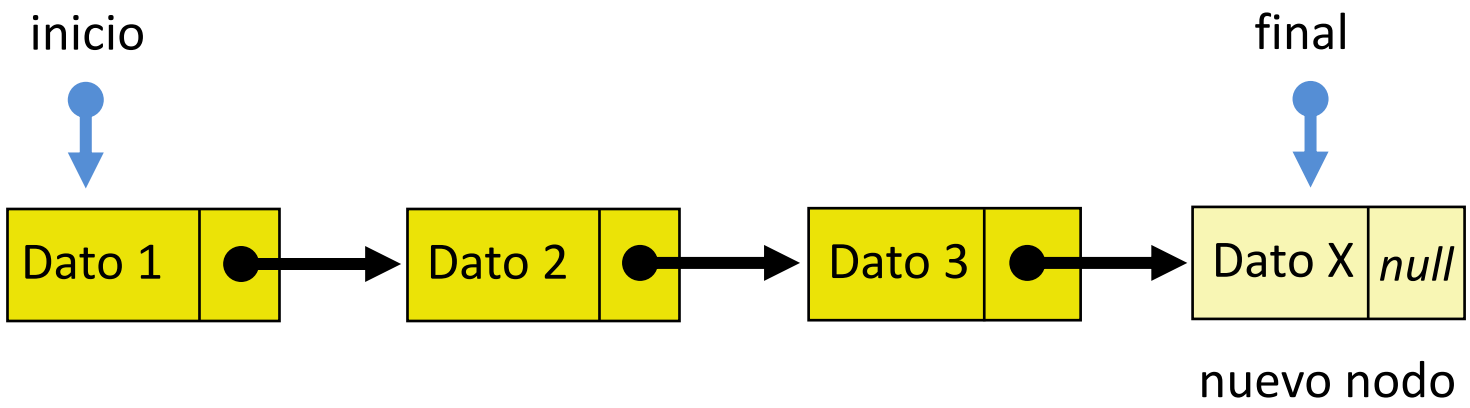


Cola

Usando listas enlazadas:

- La adición es por el final de la lista y la eliminación es por el inicio de la lista

*Al adicionar Dato X, enlazamos un nodo al nodo **final***

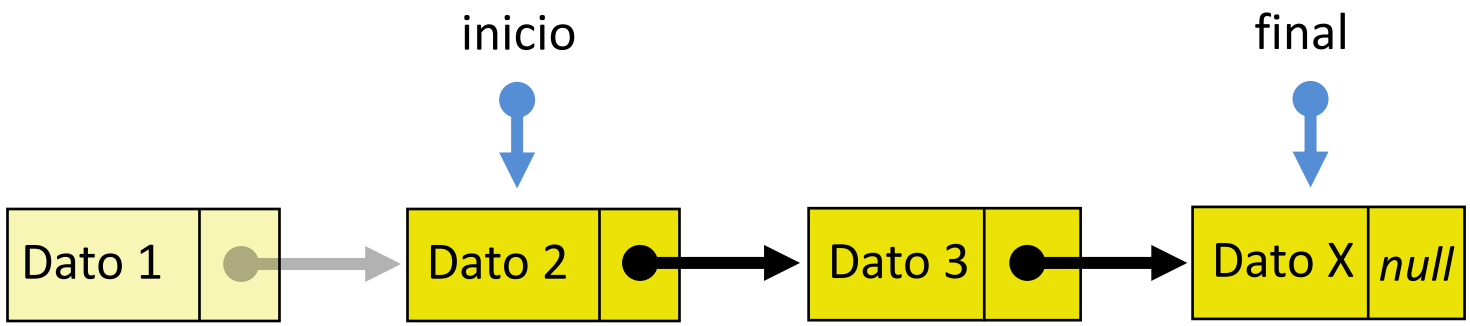


Cola

Usando listas enlazadas:

- La adición es por el final de la lista y la eliminación es por el inicio de la lista

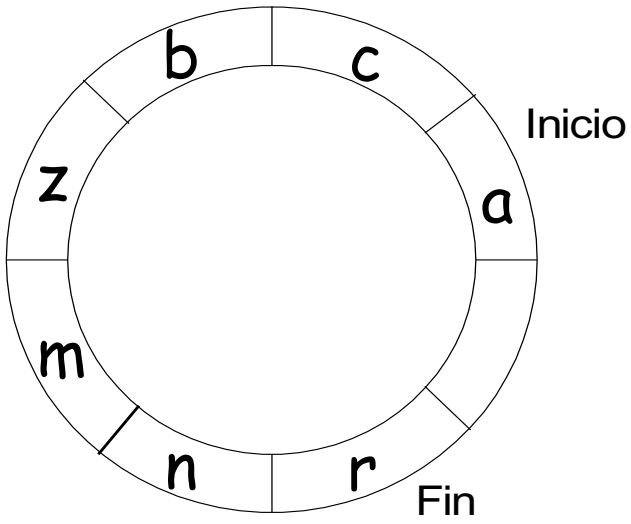
*Al eliminar, cambiamos el puntero **inicio** al siguiente nodo: Dato 2*



Cola

Cola Circular (cíclica)

- Es una cola en la que se optimiza el uso de los espacios.
- Es una estructura en donde el Inicio y el Final de la cola se desplazan en la estructura de acuerdo a las inclusiones y eliminaciones.



Cola

Cola doble o bicola

- Son colas en donde los datos se pueden añadir y quitar por ambos extremos.
- También se les llama DEQUE (Double Ended Queue).
- Una forma de representación es con arreglos con Inicio y Fin apuntando a cada uno de los extremos.

Cola

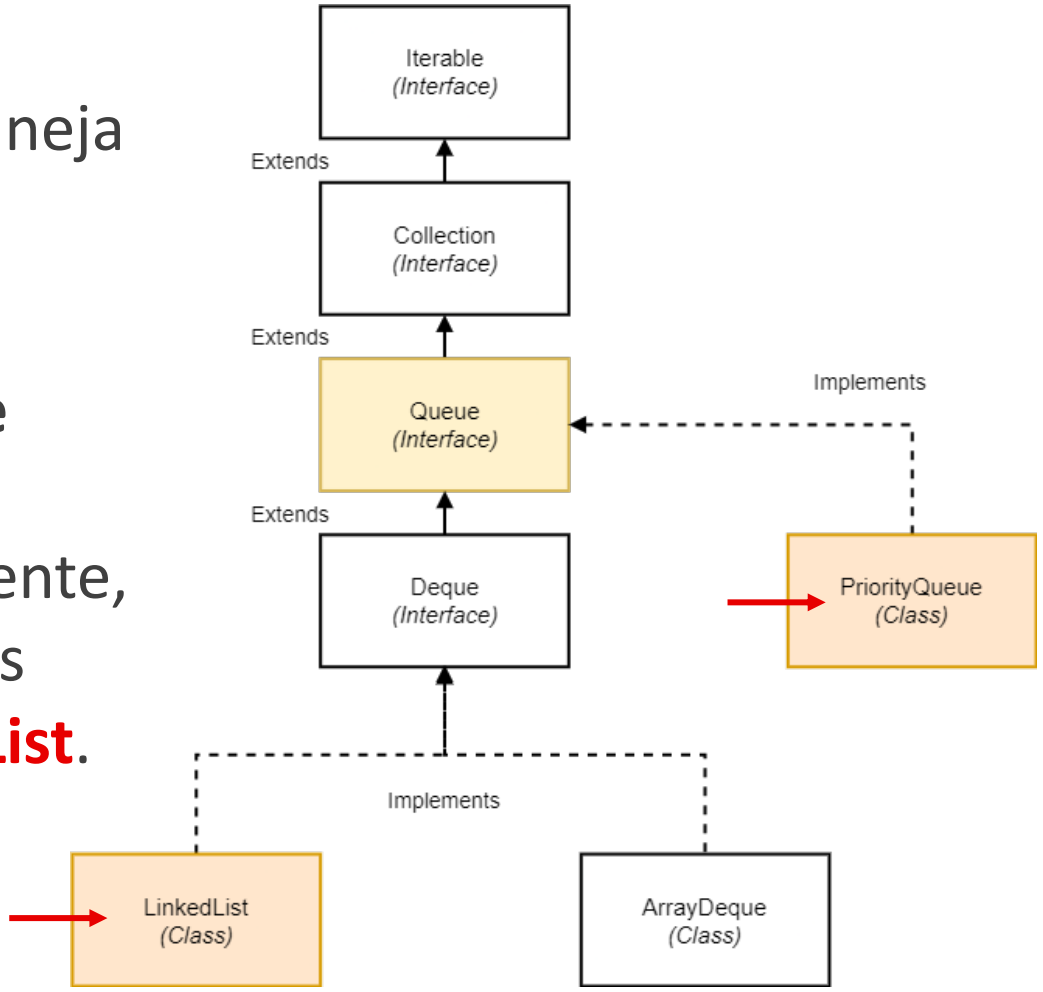
Cola con prioridades

- En este tipo de colas, cada elemento tiene asociado una prioridad, basada en un criterio objetivo.
- Se puede implementar usando un solo arreglo que almacene los diferentes elementos con prioridades.
- También se puede implementar usando varios arreglos. Un arreglo para los elementos de la misma prioridad.

Implementación

En Java tenemos una interface **Queue** que maneja el principio **FIFO**

Para trabajar con **Queue** necesitamos una clase concreta que lo implemente, siendo las mas populares **PriorityQueue** y **LinkedList**.



Implementación

Cola usando LinkedList en Java

Crear

1. Importar las librerías requeridas:

```
import java.util.LinkedList;
import java.util.Queue;
```

2. Crear un nuevo Queue

```
// Crear
Queue<String> cola = new LinkedList<>();
```

Implementación

Cola usando LinkedList en Java

isEmpty

```
// (isEmpty) Esta vacia
boolean estaVacia = cola.isEmpty();
System.out.println(estaVacia);
```

Implementación

Cola usando LinkedList en Java

Enqueue

```
// (Enqueue) Agregar nuevos elementos
cola.add("Dato 1");
cola.add("Dato 2");
cola.add("Dato 3");
cola.add("Dato X");
```

Implementación

Cola usando LinkedList en Java

Dequeue

```
// (Dequeue) Eliminar elemento en inicio
cola.remove();
```

Implementación

Cola usando LinkedList en Java

First/Peek

```
// (First) Examina elemento en inicio
System.out.println(cola.peek());
```

Práctica

Ejercicio 1:

Desarrolle un pequeño programa que permita administrar las colas en el proceso de vacunación*. Realizar lo siguiente:

- Un método que reciba una cola y un nombre a consultar. El método deberá retornar Verdadero/Falso si el nombre en consulta corresponde a la siguiente persona en vacunarse.
- Un método que reciba una cola y ejecute una “atención”. Es decir, el método deberá remover al siguiente elemento en la cola e imprimir:
  - “Vacunación exitosa para <nombre>”

* Considere una cola simple

Conclusiones

- Las colas permiten mantener un orden de atención, evitando conglomeraciones.
- Las colas se implementan usando arreglos o con listas enlazadas.
- Recordar que las colas se presentan en cualquier situación de atención a clientes o de requerimiento de algún servicio, por lo que su implementación en una organización permite mejorar los tiempos de atención a los clientes.
- Tener presente que las colas con prioridades, permiten implementar situaciones como atención a personas con discapacidad o con urgencia de atención.

Referencias bibliográficas

- **CAIRO O. (2013)** Estructuras de Datos. México: Mc Graw Hill.

GRACIAS