

UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO
Facultad de Ciencias



Estructuras de Datos

Práctica 11: Colas

Profesora:

Amparo López Gaona

Ayudante: Adrián Aguilera Moreno

Ayudante de Laboratorio: Kevin Jair Torres Valencia

Objetivos

- Comprender el funcionamiento y las características fundamentales de la estructura de datos Cola.
- Implementar una estructura de datos tipo Cola utilizando una interfaz separada de su implementación.
- Extender la funcionalidad básica de una Cola con operaciones avanzadas.

Introducción

Una Cola (Queue) es una estructura de datos lineal que almacena una colección de elementos sin restricción en cuanto al valor de los mismos, con o sin repetición, sin embargo el acceso a ellos está restringido al permitir que se inserten siempre por el mismo lado (denominado final o parte trasera de la cola) y se eliminen por otro (denominado frente de la cola). Siguen el principio *FIFO* (*First In, First Out*), es decir, el primer elemento que se agrega a la cola será el primero en ser eliminado. Esta estructura se asemeja a una fila de personas esperando ser atendidas: la primera persona que llega es la primera en ser atendida.

Las colas son ampliamente utilizadas en Ciencias de la Computación y tienen numerosas aplicaciones como en sistemas operativos (gestión de procesos), redes (gestión de paquetes), simulaciones, entre otros contextos donde el orden de llegada es importante para determinar el orden de procesamiento.

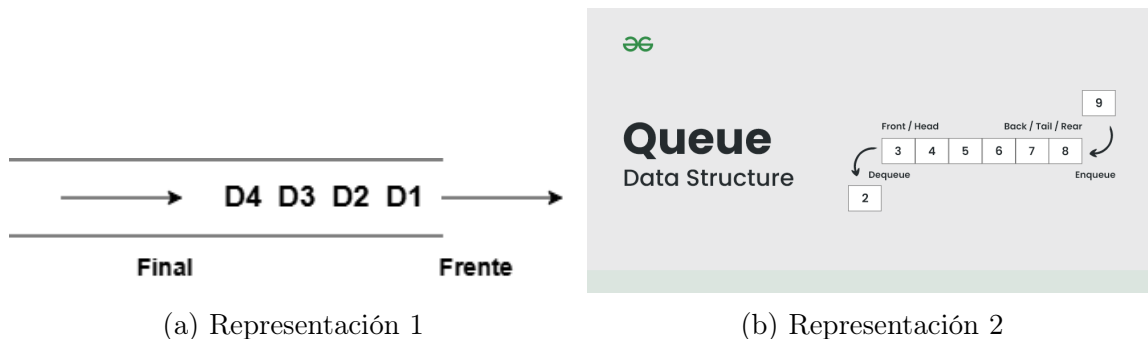


Figure 1: Cola de Datos

Operaciones básicas de una Cola

Una Cola típicamente soporta las siguientes operaciones básicas:

- **Encolar (Enqueue)**: Añade un elemento al final de la cola.
- **Desencolar (Dequeue)**: Elimina y devuelve el elemento del frente de la cola.
- **Peek/Front**: Observa el elemento en el frente de la cola sin eliminarlo.
- **IsEmpty**: Verifica si la cola está vacía.
- **Size**: Devuelve el número de elementos en la cola.

Representación en memoria

Existen diferentes formas de implementar una Cola:

1. Implementación basada en arreglos: Utiliza un arreglo para almacenar los elementos.
2. Implementación basada en listas enlazadas: Utiliza nodos enlazados para almacenar los elementos.

En esta práctica, utilizaremos la implementación basada en listas enlazadas, donde cada elemento se representa como un nodo que contiene el dato y una referencia al siguiente nodo. Mantendremos referencias tanto al inicio como al final de la cola para permitir operaciones eficientes de encolar y desencolar.

Desarrollo

En esta práctica, deberán completar la implementación de una Cola genérica, siguiendo el principio de separación de interfaz e implementación. Se te proporcionarán algunos archivos base que deberás extender.

Para ello tendrán que implementar los métodos:

- **Constructor:** Crear la cola vacía.
- **Encolar:** Agrega un elemento de tipo *T* a la cola
- **Desencolar:** Elimina el elemento del inicio de la cola, si la cola no es vacía.
- **Peek:** Devuelve el dato contenido en el inicio de la cola, si la cola no es vacía.
- **esVacía:** Verifica si la cola es vacía.
- **Longitud:** Devuelve el número de elementos de la cola.
- **Clonar:** Crea una copia exacta de la cola actual sin modificar la original.
- **Combinar:** Une otra cola a la cola actual, manteniendo el orden *FIFO*.
- **Invertir:** Inverte el orden de los elementos en la cola.

Archivos proporcionados

- **InterfazCola:** Interfaz que define el comportamiento de una estructura de datos tipo Cola.
- **ColaException:** Clase de excepción personalizada para las operaciones de Cola.
- **ColaImp:** Implementación parcial de la interfaz Cola.
- **Main:** Clase para probar la implementación.

Formato de Entrega

1. Las prácticas se entregarán en parejas.
2. NO incluir los archivos .class dentro de la carpeta.
3. Los archivos de código fuente deben estar documentados.
4. Se pueden discutir y resolver dudas entre los integrantes del grupo. Pero cualquier práctica plagiada total o parcialmente será penalizada con cero para los involucrados.
5. La práctica se debe subir al Github Classroom correspondiente.
6. La entrega en classroom debe contener el link HTTPS y SSH de su repositorio y es lo único que se debe entregar.
7. El horario y día de entrega se acordará en la clase de laboratorio y no deberá sobrepasar 2 clases de laboratorio.