Estructuras de Datos

TDA Cola

Gonzalo Gabriel Méndez, Ph.D.

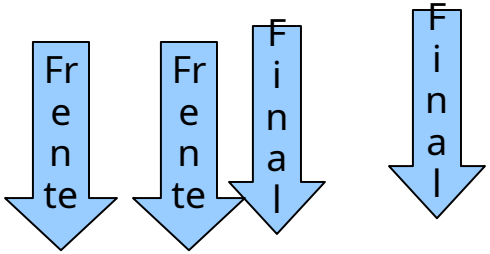
www.ggmendez.com

Facultad de Ingeniería en Electricidad y Computación

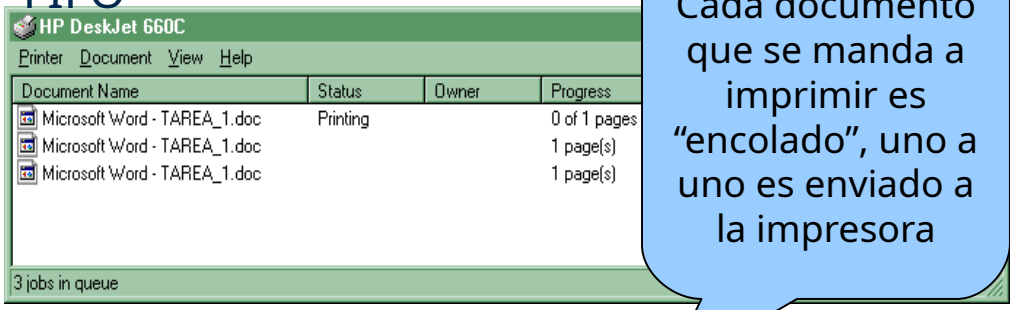
ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
ESPOL

DEFINICION

- Abunda este concepto, en la vida cotidiana
  - Cuando vamos al cine, para comprar las entradas
  - Cuando estamos en el supermercado, en el banco, etc.



- Como funciona
  - Se puede decir que la cola tiene 2 extremos:
    - FRENTE, Y FINAL
  - Todo el que llega se ubica al final de la cola
    - Todo el que sale, lo hace por el frente de la cola
  - La cola es por turno
    - El primero en llegar, tiene la seguridad de que será el primero en salir:
      - FIRST IN FIRST OUT -> FIFO
- La computadora las utiliza:
  - Para manejar la impresión de documentos, tiempo compartido, etc.



Queue -> Cola
Cada documento que se manda a imprimir es “encolado”, uno a uno es enviado a la impresora

OPERACIONES BASICAS

- Al suponer que existe un TDA Cola, podemos:
  - Cola Q;
- Todo TDA presenta operaciones básicas, en este caso:
  - **EnColar**
    - *Insertar un elemento nuevo a la cola, al final de la misma,*
    - *El final aumenta*
  - **DesEnColar**
    - *Cuando un elemento es removido de la cola*
    - *Remueve el elemento del **frente***
    - *Retorna el elemento removido*
    - *No se puede ejecutar si la cola **EstaVacia***
- Así como en la pila
  - Cualquier intento de acceder a elementos en una Cola Vacía:
    - *SUBDESBORDAMIENTO DE LA COLA*

Interface Queue<E>

Type Parameters:

`E - the type of elements held in this collection`

Interface Queue<E>

Type Parameters:

E - the type of elements held in this collection

All Known Implementing Classes:

AbstractQueue, ArrayBlockingQueue, ArrayDeque, ConcurrentLinkedDeque, ConcurrentLinkedQueue, DelayQueue, LinkedBlockingDeque, LinkedBlockingQueue, LinkedList, LinkedTransferQueue, PriorityBlockingQueue, PriorityQueue, SynchronousQueue

Interface Queue<E>

Type Parameters:

`E - the type of elements held in this collection`

All Known Implementing Classes:

AbstractQueue, ArrayBlockingQueue, ArrayDeque, ConcurrentLinkedDeque, ConcurrentLinkedQueue, DelayQueue, LinkedBlockingDeque, LinkedBlockingQueue, LinkedList, LinkedTransferQueue, PriorityBlockingQueue, PriorityQueue, SynchronousQueue

Métodos para añadir y remover elementos

	Throws exception	*Returns special value*
Insert	`add(e)`	`offer(e)`
Remove	`remove()`	`poll()`
Examine	`element()`	`peek()`

Colas de Prioridad

TDA COLAS DE PRIORIDAD

- En las colas normales
  - Las operaciones están definidas en función del orden de llegada de los elementos
    - *Al encolar un elemento ingresa al final de la cola*
    - *Al desencolar, sale del frente de la cola*
  - En una cola, los elementos esperan por ser atendidos
    - *Es justo, porque el que llega primero, se atiende primero*
- En una cola de prioridad
  - Prioridad
    - *El orden de atención, no esta dado solo por el orden de llegada*
    - *Cada elemento, tendrá asociado una cierta prioridad*
    - *Cada elemento será "procesado", según su prioridad*

TIPOS DE COLAS DE PRIORIDAD

- Hay dos tipos de colas de prioridad
  - De Prioridad Ascendente
    - *EnColar: son encolados arbitrariamente(PQEnColar)*
    - *DesEnColar: se remueve el elemento mas pequeño de la cola(PQMinDesEncolar)*
  - De Prioridad Descendente
    - *EnColar: son encolados arbitrariamente*
    - *DesEnColar: se remueve el elemento mas grande de la cola(PQMaxDesEncolar)*
- Las colas de prioridad pueden contener
  - Enteros, Reales
  - Estructuras,
    - *Estarían ordenadas en base a uno o mas campos*

DESENCOLAR EN COLAS DE PRIORIDAD

- Al **encolar** un elemento en este tipo de cola
  - Se encola al final de los elementos *con la misma prioridad*
- El **desencolar** elementos de una cola
  - Quiere decir, que ese elemento es escogido para ser “atendido”
  - Se elige **el primer elemento** con la **mayor/menor prioridad**
  - En las de prioridad ascendente, por ejemplo
    - *Se busca atender primero al de menor valor en toda la cola: BUSCAR*
    - *Y luego sacarlo*
- Es decir, existe un conjunto de prioridades
  - Cada prioridad tendrá un conjunto de elementos que se comportara como una cola