# TDA Administrador de Colas prioridad

El administrador de colas tiene como característica principal que cada cola está identificada por una clave que no puede repetirse. Los elementos también son claves únicas no pueden repetirse ni dentro, ni fuera de la cola. Es decir, el elemento 1, solo puede estar en una de las colas.

A continuación, vamos a listar las operaciones, junto a sus comportamientos, que nos permite especificar el TDA :

1. **Inicializar:** Dada una cantidad N de puestos de atención, crea dicha cantidad de puestos, numerándolos de 1 a N (constituyéndose identificadores) e inicializa el sumador de tiempo de espera de atención de los puestos en 0; ya que se encuentran “libres” por el momento.
2. **Acolar:** Dado el tiempo de atención estimado en minutos, el Administrador de Colas:
   1. Le asigna un identificador
   2. lo agrega a la cola del puesto de atención de menor demora
   3. Suma la demora estimada del elemento a la del puesto. Es decir, si la cola tenía demora 30 y el elemento tiene un tiempo de epera estimada de 10’, el tiempo de espera del puesto será ahora es de 40’= 30’+10. Este elemento no puede ser vuelto a agregar a menos que se lo elimine primero, ya que el mismo elemento no podría existir en otra cola.
   4. Devuelve el identificador del elemento

Para poder agregar un elemento el TDA debe estar inicializada.

1. **Desacolar:** Desacola el próximo elemento a ser atendido. Decrementa al tiempo de espera del puesto correspondiente, el tiempo de atención del elemento. Por ejemplo, si el puesto tenía un tiempo de espera estimada de 30’ y el elemento un tiempo estimado de 10’, ahora la cola tiene una demora estimada de 20’=30’-10’.
2. **Programación:** devuelve una cola prioridad, donde la prioridad es el tiempo estimado de atención de los elementos a partir de la apertura de los puestos de atención. Se supone que todos los puestos abren al mismo tiempo. El TDA debe estar inicializado. NOTA: En esta cola prioridad, la prioridad mayor es la del elemento de menor tiempo estimado de atención. Es decir, el elemento con tiempo estimado de atención 0’ va primero que el de tiempo de atención 30’.
3. **CantidadDeColas:** Devuelve la cantidad de colas con que se inicializó el administrador de colas prioridad. Se supone que el TDA está inicializado.
4. **Primero** Devuelve el id del próximo elemento a ser llamado. Se supone que el TDA está inicializado.
5. **Tiempo estimado:** Devuelve el tiempo estimado en que va a ser llamado el próximo elemento, desde la apertura de los puestos
6. **Puesto del próximo Elemento:** Devuelve el id del puesto del próximo elemento a ser atendido. Se supone que el TDA está inicializado y que hay elementos sin ser atendidos.
7. **Puesto del elemento**: Dado un elemento devuelve el puesto de atención en el que se encuentra. Se supone que el TDA está inicializado y el elemento se encuentra.
8. **Elementos:** Devuelve un diccionario con los elementos encolados y la demora estimada de atención**.**

public interface AdministradorDeColas {

void inicializar( int cantidad);

// siempre que el TDA esté inicializado

void acolar(int idElemento, int TiempoEstimado);

// siempre que el TDA esté inicializado

void desacolar();

// siempre que el TDA esté inicializado

ColaPrioridadTDA programacion ();

// siempre que el TDA esté inicializado

ColaTDA ColaTDA colas();

// siempre que el TDA esté inicializado

int cantColas();

// siempre que el TDA esté inicializado

int primero();

// siempre que el TDA esté inicializado y cola no vacías

int estimado();

// siempre que el TDA esté inicializado y cola no vacías

int puestoProximoElem( int idElemento);

// siempre que el TDA esté inicializado y cola no vacías

int puestoDelElem( int idElemento);

// siempre que el TDA esté inicializado y cola no vacías

DiccionarioSimpleTDA elementos();

// siempre que el TDA esté inicializado y cola no vacías

}

# IDENTIFICAR TDAS PARA:

El estudiante debe encontrar el TDA adecuado para representar las siguientes relaciones entre los datos ( Le vamos consultando para que ellos lo saquen)

|  |  |
| --- | --- |
| DESCRIPCION | TDA |
| Puesto relación entre IdPuesto de Atención con tiempo de atención | **Diccionario Simple** |
| Puesto en que se encuentra el elemento: idElemento, idPuesto de Atención | **Diccionario Simple** |
| idElemento, tiempo de atención del elemento | **Diccionario Simple** |
| Programación (idElemento, tiempo estimado de atención desde apertura de los puestos) ordenado por menor tiempo. | **Cola Prioridad** |

# Enunciados de TPs:

# Programa para gestionar múltiples turnos del banco.

Desarrollar un programa para gestionar los turnos para atención del banco MyBank. Para esto, se propone una atención donde se irá llamando por turnos, para esto se tendrán múltiples puestos de atención:

Demoras estimadas en función al tipo servicio solicitado

* clientes. 40’
* Fila no clientes: 10’
* Fila Jubilados. 30’
* Fila Servicios.
  + Solicitud de Plazo Fijo=10’
  + Solicitud de créditos hipotecarios:40’
  + Solicitud de créditos:15

Nomenclaturas:

1. Fila Jubilados. Con la nomenclatura de letra asignada al turno “J”.
2. Fila Servicios. Plazo fijo, créditos hipotecarios y créditos generales. Para esto, la nomenclatura de servicios será “S” y la nomenclatura para cada uno de los servicios será:
   1. “PF” para Plazos Fijos.
   2. “CH” para créditos hipotecarios.
   3. “CG” para créditos generales.
3. Fila Clientes. Con la nomenclatura de letra asignada al turno “C”.
4. Fila No Clientes. Con la nomenclatura de letra asignada al turno “P”.

El programa debe poder:

* ingresar la cantidad de puestos de atención
* Ingresar las solicitudes de turnos (nomenclatura)
* Emitir la programación correspondiente de los turnos indicando :
  + Ticket (identificador del turno)
  + Demora de atención estimada a partir de la apertura del comercio.
  + Puesto en el que se atenderá.
* Realizar el test con 6 puestos de atención y 40 clientes de distinto tipo

1. Simulador Round Robin

El algoritmo de planificación de procesos, conocido como Round Robin, está diseñado especialmente para repartir los procesos entre distintos recursos, como ser servidores, procesadores, etc...

El tiempo estimado de espera del servidor surge de la suma de los tiempos estimados de cada proceso. La planificación se estructura como una cola por prioridad, donde la prioridad es el tiempo estimado en el que se disparará el proceso. Cuanto menor el tiempo estimado, mayor la prioridad del proceso.

Para implementar el Round Robin, el programa debe permitir:

1. Inicializar los servidores que atenderán los procesos, que serán numerados arbitrariamente con números que van de 1 a N
2. Ingresar los procesos (Quantum). El sistema devuelve el identificador del proceso en el sistema operativo.
3. El proceso se asignará al servidor se libere más pronto. Por ejemplo, si un servidor terminara de ejecutar su último proceso en 30’ (tiempo estimado de espera) y otro dentro de 40’, se asignará a servidor con tiempo de espera de 30’.
4. Programar los procesos, indicando servidor, hora de ejecución y servidor en el que se ejecutó.
5. Emitir un listado indicando:
   * Proceso
   * Servidor
   * Hora estimada de ejecución
6. Realizar la simulación con al menos 6 servidores y 40 procesos

# Programa para gestionar turnos en una farmacia.

Desarrollar un programa para gestionar los turnos para atención de la Farmacia FarmaPolis que dispone de 6 puestos de atencíon. Para esto, se propone una atención donde se irá llamando por turnos, para los que se tendrán múltiples puestos de atención:

Demoras estimadas en función al tipo servicio solicitado

* Obra Social. 30’
* Particular: 20’
* PAMI.
  + + de 3 Recetas. 50’
  + 2 o 3 Receta: 30’
  + 1 Receta 10’

Nomenclaturas:

1. Obra Social: Con la nomenclatura de letra asignada al turno “O”.
2. Particular: Con la nomenclatura de letra asignada al turno “P”.
3. PAMI
   1. “P+3”: + de 3 Recetas.
   2. “P3” :2 o 3 Receta
   3. “P1” :1 Receta.

El programa debe poder:

* ingresar la cantidad de puestos de atención
* Ingresar las solicitudes de turnos (nomenclatura)
* Emitir la programación correspondiente de los turnos indicando :
  + Ticket (identificador del turno)
  + Demora de atención estimada a partir de la apertura del comercio.
  + Puesto en el que se atenderá.
* Realizar el test con 6 puestos de atención y 40 clientes de distinto tipo