

### Hoja de trabajo No. 5

**Realizar:** Programa de implementación de colas con prioridad.

**Realizarse:** en forma **INDIVIDUAL**.

**Fecha de entrega:** miércoles 2 de septiembre

**Objetivos:**

- Implementación de las operaciones para manejo de listas con prioridad, utilizando Heaps.
- Utilización de la implementación que el JCF (Java Collection Framework) proporciona para el manejo de las colas con prioridad.
- Utilización de un profiler para ver el uso de los recursos (CPU, memoria) de un programa.

**Programa a realizar:**

Debe implementar una cola con prioridad:

- Definir una interfaz genérica para colas con prioridad. Por lo menos debe tener los métodos de agregar, retirar un elemento y verificar si está vacía la cola.
- Definir una clase que implemente esta interfaz utilizando Heaps.
- La clase genérica debe implementar la interfaz `Comparator`.
- Los métodos de la interfaz `Comparator` (`compare` y `equals`) servirán para indicar el orden de la cola con prioridad, de tal forma que X tiene mayor prioridad que Y y que por lo tanto es retirado más rápidamente de la cola, si  $X > Y$  (ejemplo:  $100 > 50$ , por lo que 100 sale antes de la cola que 50). Note que este es un orden inverso al *orden natural* que usualmente se define (lo que vimos en la clase es que se usa que  $50 < 100$ , por lo que 50 sale antes de la cola que 100).

Su programa genera al azar mil (1,000) números enteros y los almacena en cualquier colección del JCF que usted desee. Debe mostrar en pantalla el valor más pequeño y el más grande que se generó.

Con esos números generados se debe:

- Parte A:
  - Guardarlos en la cola con prioridad que usted implementó (recuerde utilizar `Comparator` que usted desarrolló para determinar el orden de la cola).
  - Retirarlos de la cola con prioridad. Debe mostrar el primer y último valor que se retira de la cola con prioridad.
- Parte B:
  - Utilizar la interfaz e implementación proporcionadas por el JCF de `PriorityQueues`, utilizando el constructor que permite indicar el `Comparator` para el orden de la cola.
  - Guarde y retire los mismos números generados al azar en ese `PriorityQueue`. Muestre el primer y último elemento retirado.

Utilizar el profiler para medir el uso de CPU y Memoria de su programa en cada una de las Partes A (su implementación) y B (implementación con el JCF).

**Tareas:**

- Construir el programa que use la implementación de Colas con Prioridad desarrollado por usted y la implementación proporcionada por el JCF.
- Elaborar un video con la operación de su programa. Colóquelo en un sitio externo al Sakai y solo mande su enlace. El video debe mostrar el uso del profiler e identificar claramente las estadísticas que se obtienen en la parte A y B de su programa.

Debe subir a Sakai su programa y el enlace al video producido.



**Calificación:** su programa debe funcionar para ser calificado.

Aspecto	Puntos
Implementación de una cola con prioridad usando Heap.	30
Utilización del JCF para la cola con prioridad.	25
Uso del profiler para ver la utilización de CPU y memoria	25
Video que muestre el funcionamiento del profiler	20
<b>TOTAL:</b>	<b>100</b>