

Trabajo Práctico Nº 8

Tema: Programación concurrente: Executor Service

Fecha Inicio: 19/10/2023 **Fecha de Entrega:** 02/11/2023

Actividades:

- 1) El Executor Framework permite crear un Pool de ejecución de hilos (ahora llamados tareas/tasks). Revise el ejemplo “EjemploPool” subido al aula virtual. Implemente un Pool de ejecución tamaño 3 para ejecutar 10 tareas, donde cada una de ellas realizará el cálculo siguiente, recibiendo como parámetro (root) el entero correspondiente al número de tarea:

```
public static SumRootN(int root)
{
    double result = 0;
    for (int i = 0; i < 10000000; i++)
    {
        result += Math.exp(Math.Log(i) / root);
    }
    System.out.println("Resultado" + result);
}
```

- 2) Implemente una clase que muestre la fecha y hora actual del sistema. Desde el main() lance la ejecución de dicha tarea cada 2 segundos de forma ininterrumpida, para ello deberá programar la ejecución del ExecutorService:

```
ScheduledExecutorService scheduledExecutorService =
Executors.newScheduledThreadPool(1);
```

```
scheduledExecutorService.scheduleAtFixedRate(tarea a ejecutar, espera inicial,
espera, unidad de tiempo utilizada);
```

- 3) Simule la atención en un supermercado donde existen 3 cajas. Cada cajero demora en atender a un cliente un tiempo variable entre 1” y 3”. Suponga que la cola de espera es de 50 clientes. Deberá indicar por pantalla el Número de cliente que es atendido, cuando comienza la atención y cuando finaliza la misma.
- 4) Programe la ejecución de una tarea repetitiva, que cada 5 segundos chequee si en un directorio (carpeta/folder) a elección se ha creado un nuevo archivo. Para ello deberá iniciar la lectura y carga de los nombres de archivos para luego verificar si ha agregado uno nuevo, en caso de que un nuevo archivo sea creado se deberá mostrar por pantalla “Nuevo archivo [nombre_archivo], con tamaño [tamaño_archivo]”. Deberá copiar un nuevo archivo a dicha carpeta para comprobar el proceso.

- 5) Se dispone del siguiente arreglo:

```
long[] vector = new long[] { 100477L, 105477L, 112986L, 100078L,
165987L, 142578L };
```

Se desea hacer el siguiente cálculo de cada número con la siguiente función que demora varios segundos:

```
static BigInteger M = new BigInteger("1999");
```

```
private static BigInteger compute(long n) {  
    String s = "";  
    for (long i = 0; i < n; i++) {  
        s = s + n;  
    }  
    return new BigInteger(s.toString()).mod(M);  
}
```

Para ello deberá definir un Pool de ejecución de tamaño 2 para ejecutar dichos cálculos.

NOTA

Los puntos obligatorios de este TP y que serán expuestos por los grupos (que en la próxima clase se sortearán) son: Punto 3, Punto 4 y Punto 5.