

TRABAJO SEMANA 7

Nombres: Verónica Luna, María Montaña, Nahomi Cabrera

Fecha: 23 de mayo de 24

Líneas explicadas

```
var pos = 0;
int a = 5;
int[][] mat = ejercicios.sumaFilasTotal.llenarMatriz( f: 12, c: 12);
int[] sums = new int[mat.length];

System.out.println("Longitud: " + mat.length);
```

En esa parte del código lo único que se hace es crear variables. Primero se instancia la posición de cada una de las sumas de cada fila, por ejemplo, en este caso son 12 filas, entonces el vector sums será de longitud 12, y las posiciones serán 0 1 2 3 4... 11.

Y se imprime la longitud.

```
var numCores = Runtime.getRuntime().availableProcessors();
```

```
10 numCores = 8
01
```

En esa línea se guarda el número de procesadores disponibles de mi máquina.

```
ThreadPoolExecutor executor = (ThreadPoolExecutor) Executors.newFixedThreadPool(numCores);
```

Aquí se hace un cast.

Aquí se crean ocho hilos reutilizables, porque recibe el número de procesadores del computador.

```
CountDownLatch endController = new CountDownLatch(12);
```

Se comporta casi como un join, y se pasa el número de filas de la matriz.

```

for(var row: mat) {
    Thread thread = new Thread(new SumRowTask(row, pos ++, sums, endController, a));
    executor.execute(thread);
    System.out.print(a + " - ");
}
System.out.println("\n");

```

Este ciclo se encarga de crear hilos y ejecutarlos además de imprimir la variable a, que en este caso imprimir 5. Supongo que para demostrar que a no es la misma a que se envía a la clase SumRowTask.

```

try {
    endController.await();
} catch (InterruptedException e) {
    throw new RuntimeException(e);
}

```

Es para controlar que el contador (12) haya llegado a cero. Para que se entienda mejor, el main se detiene, espera a todos los hilos y de ahí sigue ejecutándose.

```

System.out.println("\n>>> FIN <<<\n");
executor.shutdown();

```

En esta línea apaga a los hilos, siempre y cuando hayan acabado todas sus tareas.

```

var sumTotal = 0;
ejercicios.sumaFilasTotalHilos2.printRow(sums); // ci
System.out.println("\n\n\n");

for(var sum: sums) {
    sumTotal += sum;
}

System.out.printf("Suma total: %d\n", sumTotal);
ejercicios.sumaFilasTotalHilos2.printRow(sums);
System.out.println();
ejercicios.sumaFilasTotalHilos2.presentarMatriz(mat);

```

Y este conjunto de líneas se imprimen los valores de las sumas de cada una de las filas. Para finalmente obtener todos estos valores, y sumarlos y presentar la suma total de la matriz.

Lo que se presenta en pantalla:

```

Longitud: 12
>>> PRESENTAR MATRIZ <<<
1 9 7 3 7 5 8 3 6 4 7 2
9 5 1 5 1 5 1 4 6 8 1 1
6 4 2 6 5 5 3 9 4 6 5 7
5 7 8 8 7 4 7 8 7 8 6 7
2 4 8 3 8 5 7 9 4 5 1 9
6 9 2 5 5 6 3 9 5 9 5 8
1 5 9 1 8 2 9 1 5 3 3 2
8 2 9 3 7 3 1 3 7 3 3 6
1 4 7 4 1 1 6 2 9 3 5 1
3 2 1 7 2 9 8 8 8 4 5 9
9 2 2 1 4 8 8 8 5 2 1 8
5 1 9 9 7 3 2 7 4 5 9 4

```

```
>>> Creando y ejecutando hilos <<<  
5 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - 5 - 5
```

```
Hilo -----> 26  
Hilo -----> 24  
Hilo -----> 26  
Hilo -----> 26  
Hilo -----> 26  
Hilo -----> 28  
Hilo -----> 38  
Hilo -----> 30  
Hilo -----> 32  
Hilo -----> 34  
Hilo -----> 36  
Hilo -----> 24
```

```
>>> FIN <<<
```

```
62 47 62 82 65 72 49 55 44 66 58 65
```

Suma total: 727

62 47 62 82 65 72 49 55 44 66 58 65

>>> PRESENTAR MATRIZ <<<

1 9 7 3 7 5 8 3 6 4 7 2

9 5 1 5 1 5 1 4 6 8 1 1

6 4 2 6 5 5 3 9 4 6 5 7

5 7 8 8 7 4 7 8 7 8 6 7

2 4 8 3 8 5 7 9 4 5 1 9

6 9 2 5 5 6 3 9 5 9 5 8

1 5 9 1 8 2 9 1 5 3 3 2

8 2 9 3 7 3 1 3 7 3 3 6

1 4 7 4 1 1 6 2 9 3 5 1

3 2 1 7 2 9 8 8 8 4 5 9

9 2 2 1 4 8 8 8 5 2 1 8

5 1 9 9 7 3 2 7 4 5 9 4

Process finished with exit code 0