

RA1- CE G y H**Suma de ficheros de contabilidad****TABLA DE CONTENIDO**

ENUNCIADO.....	2
ENTREGA	4
EVALUACIÓN	5

ENUNCIADO

Vamos a aprovechar el paralelismo del ordenador para calcular la suma total de n ficheros de números. Estos ficheros se llamarán `contabilidad1.txt`, `contabilidad2.txt` y `contabilidad3.txt`, etc. Estos ficheros tienen unas cantidades de números, uno en cada línea. Por ejemplo, el fichero `contabilidad1.txt` podría ser:

```
1
54
78
68
48
```

Lo mismo con los otros ficheros.

- Proyecto **GeneraFichero**: Para generar los ficheros, crea un proyecto de **Netbeans** que genere un fichero con n números aleatorios. Este proyecto servirá para crear los ficheros de contabilidad de prueba que necesitaremos en los siguientes proyectos. Este proyecto no habrá que entregarlo.

Ahora, tendrás que hacer los siguientes programas en diferentes proyectos:

- Proyecto **Suma**: Lee un fichero cuyo nombre es pasado *como argumento* (**`String[] args`**) y calcula la suma total de los números de ese fichero. El resultado de la suma total lo guardará en otro fichero con el mismo nombre, pero acabado en `.res`. Por ejemplo: `contabilidad1.txt.res`. Además, imprimirá el resultado por pantalla (salida estándar). De este proyecto deberás crear un `.jar` llamado `Suma.jar`.
 - Ejecución: `java -jar Suma.jar contabilidad1.txt`

- Proyecto **SumaTotales**: Lanza *n* procesos de *Suma.jar* pasándole como argumento los nombres de los ficheros de contabilidad. Tendrá que lanzar un proceso ejecutando *Suma.jar* por cada fichero de contabilidad.
 - Por ejemplo, con tres ficheros la ejecución del comando sería:
 - `java -jar SumaTotales.jar contabilidad1.txt contabilidad2.txt contabilidad3.txt`
 - El proceso **SumaTotales** se deberá bloquear hasta que todos los procesos hijos terminen.
 - Prueba dos versiones:
 - El proceso **SumaTotales** ejecutará los procesos hijo de manera secuencial. Es decir, hasta que no se termine de ejecutar la suma de un fichero de contabilidad, el proceso padre se bloquea para lanzar el siguiente.
 - El proceso **SumaTotales** ejecutará los procesos hijo de forma paralela. Es decir, las sumas de los *n* ficheros de contabilidad se harán en paralelo y el proceso padre esperará a que todos los procesos hijos terminen.
 - Por último, el proceso padre deberá leer los *n* ficheros generados *.res* para sumar las sumas parciales de los *n* ficheros de contabilidad y mostrar por pantalla la suma total.

ENTREGA

- El estudiante debe adjuntar 3 archivos en un zip llamado:
 - La documentación como *pdf*.
 - Un zip con los proyectos de Netbeans **Suma** y **SumaTotales** utilizando **Maven**
 - Ficheros de contabilidad utilizados para las pruebas.
 - Los archivos **.jar** correspondientes.
- Se debe documentar el código correctamente, explicando cómo funciona con comentarios en el código.
- El *pdf* con la documentación tendrá el siguiente formato:
 - Portada (Título y nombre del estudiante).
 - Índice.
 - Cabecera.
 - Módulo y curso.
 - Pie.
 - Nombre del estudiante y número de página.
 - Fuente:
 - Calibri.
 - Tamaño:12
- Contenido
 - Portada
 - Índice
 - Demostración de la funcionalidad de la aplicación con capturas de pantalla.
 - Comparación de tiempos entre la ejecución secuencial y paralela.
Para medir tiempos de ejecución, se puede seguir este tutorial:

<https://www.techiedelight.com/es/measure-elapsed-time-execution-time-java/>
 - Conclusiones.
 - Bibliografía

EVALUACIÓN

RA1- CE-g) Se han desarrollado aplicaciones que gestionen y utilicen procesos para la ejecución de varias tareas en paralelo.

Logro	Peso %	Valoración
Se crea el proyecto Suma correctamente y se genera el .jar.	20	Sí/No
Se crea el proyecto SumaTotales correctamente en el que se lanzan los n procesos.	20	Sí/No
Los procesos se ejecutan de manera secuencial.	10	Sí/No
Los procesos se ejecutan de manera paralela.	30	Sí/No
El proceso padre espera a que todos los hijos terminen.	10	Sí/No
El proceso padre lee las sumas parciales y calcula la suma total.	10	Sí/No

RA1- CE-h) Se han depurado y documentado las aplicaciones desarrolladas.

Logro	Peso %	Valoración
Se hacen comentarios en el código representativos.	10	Sí/No
Se respeta el formato del documento pedido.	10	Sí/No
Se incluyen portada, índice, conclusiones y bibliografía.	20	Sí/No
Se hacen capturas de pantalla con demostración de la funcionalidad.	30	Sí/No
Se comparan los tiempos entre la ejecución secuencial y la ejecución paralela.	30	Sí/No