

PROYECTO 1

Total de Puntos: 70. (NO SE ACEPTAN PROYECTOS FUERA DE FECHA)

Fecha de entrega: 03/04/2019 hasta las 23:59 Hs.

Forma de entrega: Subirlo al classroom de la materia.

El objetivo del proyecto es poner en práctica los conceptos aprendidos sobre procesos, threads, lectura desde memoria y disco. Utilizando la librería Pthreads en C.

Pueden realizar la tarea en grupo de dos personas, el que quiera realizarlo solo puede sentirse libre de hacerlo. En ambos casos debe explicitarlo en el Informe adjuntado al proyecto.

La tarea principal del proyecto consiste en implementar la multiplicación de dos matrices véase el ejemplo siguiente:

$$A \cdot B = \begin{pmatrix} 2 & 0 & 1 \\ 3 & 0 & 0 \\ 5 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix} =$$
$$= \begin{pmatrix} 2 \cdot 1 + 0 \cdot 1 + 1 \cdot 1 & 2 \cdot 0 + 0 \cdot 2 + 1 \cdot 1 & 2 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 \\ 3 \cdot 1 + 0 \cdot 1 + 0 \cdot 1 & 3 \cdot 0 + 0 \cdot 2 + 0 \cdot 1 & 3 \cdot 1 + 0 \cdot 1 + 0 \cdot 0 \\ 5 \cdot 1 + 1 \cdot 1 + 1 \cdot 1 & 5 \cdot 0 + 1 \cdot 2 + 1 \cdot 1 & 5 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 \end{pmatrix} =$$
$$= \begin{pmatrix} 3 & 1 & 2 \\ 3 & 0 & 3 \\ 7 & 3 & 6 \end{pmatrix}$$

Lo primero que se debe realizar, es un programa capaz de generar matrices de números aleatorios (0 a 9) y guardarlos a un archivo que posteriormente será utilizado en la multiplicación:

La ejecución del programa debe ser:

```
>generar_matriz 1024 mat_A.txt
```

```
>generar_matriz 1024 mat_B.txt
```

Donde:

1024: es el tamaño de la matriz en este caso debe ser una matriz de 1024 x 1024.

mat_A.txt, mat_B.txt: son los nombres de los archivos respectivos.

Se deben realizar tres versiones de la solución.

- La primera debe ser resuelta utilizando un solo proceso para resolver el problema, el formato de ejecución debe ser:

```
>mult mat_A.txt mat_B.txt
```

El programa debe realizar la multiplicación de las matrices especificadas en los archivos mat_A.txt y mat_B.txt y posteriormente almacenar el resultado en un archivo resultado.txt.

Es necesario cronometrar el tiempo total (que debe ser mostrado en pantalla en segundos al completar el proceso).

- La segunda versión debe realizar la multiplicación utilizando una cantidad de threads pasada como parámetro.

El programa debe realizar la multiplicación de las matrices especificadas en los archivos mat_A.txt y mat_B.txt, utilizando la cantidad de threads pasada como tercer parámetro, se debe distribuir el trabajo de manera equitativa entre los threads especificados (desde/hasta filas y columnas que debe resolver). Posteriormente almacenar el resultado en un archivo resultado.txt.

Es necesario cronometrar el tiempo total (que debe ser mostrado en pantalla en segundos al completar el proceso).

```
>mult_pt      mat_A.txt mat_B.txt 2
```

```
>mult_pt      mat_A.txt mat_B.txt 4
```

```
>mult_pt      mat_A.txt mat_B.txt 8
```

```
>mult_pt      mat_A.txt mat_B.txt 16
```

```
>mult_pt      mat_A.txt mat_B.txt 24
```

- La tercera versión, debe asignar a cada thread, la posición inicial y final de los archivos que le corresponden en la multiplicación, distribuir el trabajo entre los threads especificados y almacenar el resultado en resultado.txt.

```
>mult_hd      mat_A.txt mat_B.txt 2
```

```
>mult_hd      mat_A.txt mat_B.txt 4
```

```
>mult_hd      mat_A.txt mat_B.txt 8
```

```
>mult_hd      mat_A.txt mat_B.txt 16
```

```
>mult_hd      mat_A.txt mat_B.txt 24
```

Es necesario cronometrar el tiempo total (que debe ser mostrado en pantalla en segundos al completar el proceso).

Posterior a la resolución de los casos realice un informe, especificando:

- Nombre de los integrantes.
- Parámetros de compilación necesarios que se deben utilizar para correr el programa.
- Alguna información adicional que le parezca importante mencionar.
- Un cuadro comparativo entre las soluciones, num. De threads, tiempo entre las 3 soluciones.
- Conclusiones a las cuales pueden llegar después de evaluar el cuadro comparativo.

Indicadores de corrección del proyecto:

- Implementación de la primera solución con un solo proceso. (10 pts.)
 - Cumple con los requisitos de flujo: entrada, salida, medición del tiempo. (2 Pts.)
 - Resultados correctos. (3 Pts.)
 - Correcta creación/manipulación de las estructuras que intervienen. (5 Pts.)
- Implementación de la segunda solución con hilos. (20 pts.)
 - Cumple con los requisitos de flujo: entrada, salida, medición del tiempo. (2 Pts.)
 - Resultados correctos. (3 Pts.)
 - Correcta creación/manipulación de las estructuras que intervienen. (5 Pts.)
 - Distribución correcta del trabajo por el número de threads especificados. (10 Pts.)

- Implementación de la tercera solución. (30 Pts.)
 - Cumple con los requisitos de flujo: entrada, salida, medición del tiempo. (2 Pts.)
 - Resultados correctos. (3 Pts.)
 - Correcta creación/manipulación de las estructuras que intervienen. (5 Pts.)
 - Distribución y funcionamiento correctos de cada thread con la lectura de los archivos. (10 Pts.)
 - Distribución correcta del trabajo por el número de threads especificados. (10 Pts.)
- Informe: (Total: 10 Pts)
 - Posee todos los datos necesarios para compilar/ejecutar las soluciones.
 - Cuadro comparativo completo con todos los resultados relevantes.
 - Conclusiones finales.