

Instrucciones¹

- Debe crear una cuenta en <https://gitlab.com> y crear un repositorio **privado** llamado **mc-8836** para las tareas del curso. Envíe un correo con su usuario de **GitLab** al profesor esteban.meneses@acm.org y al asistente alexaenz@estudiantec.cr antes del **Martes 1 de marzo a las 11:55pm**.
- Agregue a los usuarios **emeneses** (Esteban Meneses) y **sr.alexe** (Álex Sáenz) como colaborador de su repositorio **GIT** en **GitLab**. Los permisos de acceso de los usuarios **emeneses** y **sr.alexe** deben ser al menos los de un *Reporter*.
- Los fuentes para esta tarea están en el directorio **hw2** del repositorio **GIT** del profesor en **GitLab**:
`git clone https://gitlab.com/emeneses/mc-8836.git`
- Las interfaces funcionales del código fuente provisto deben respetarse.
- El único mecanismo de entrega de esta asignación es por medio de su repositorio **GIT** en **GitLab**.
- Debe crear una carpeta **hw2** en su repositorio **GIT**. Ese directorio debe contener los siguientes archivos:
 - Archivo **strassen.cpp** que contiene el código paralelo que implementa el algoritmo de Strassen.
 - Archivo **report.pdf** con un reporte resumiendo los resultados del programa (debe ser un archivo en formato PDF).
- El código paralelo debe correr en la supercomputadora **Kabré** del Centro Nacional de Alta Tecnología (CeNAT). Para poder compilar programas Cilk Plus debe ejecutar el comando:
`module load gcc/7.2.0`
También puede copiar estos comandos en el archivo `~/.bashrc` y se ejecutarán automáticamente.
- Se recomienda utilizar trabajos interactivos para el desarrollo de la tarea:
`salloc --partition=nu --time=02:00:00`
- Se debe utilizar trabajos en lote para recoger información de desempeño:
`sbatch file.slurm`
- La fecha límite de entrega de esta asignación es el **viernes 11 de marzo a las 11:55pm**. Entregas extemporáneas **no** serán aceptadas.
- La presente asignación es estrictamente **individual**.

¹Cada estudiante debe mostrar el estándar más alto de integridad académica. Presentar las ideas de otra persona sin el debido permiso es una clara violación de la política de integridad académica de la universidad y es una práctica que debilita las habilidades para resolver problemas en cualquier ámbito. Cualquier infracción a esta regla será llevada hasta sus últimas consecuencias.

Algoritmo de Strassen

Escriba un programa paralelo en Cilk Plus que implemente el algoritmo de Strassen para multiplicación de matrices. Este algoritmo calcula la multiplicación $C = AB$ de matrices cuadradas de tamaño $N \times N$ al dividir las matrices de la siguiente forma:

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \quad C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

El algoritmo luego compute las siguientes matrices intermedias:

$$M_1 = (A_{11} + A_{22})(B_{11} + B_{22})$$

$$M_2 = (A_{21} + A_{22})B_{11}$$

$$M_3 = A_{11}(B_{12} - B_{22})$$

$$M_4 = A_{22}(B_{21} - B_{11})$$

$$M_5 = (A_{11} + A_{12})B_{22}$$

$$M_6 = (A_{21} - A_{11})(B_{11} + B_{12})$$

$$M_7 = (A_{12} - A_{22})(B_{21} + B_{22})$$

Finalmente, la matriz C se construye usando los resultados intermedios:

$$C_{11} = M_1 + M_4 - M_5 + M_7$$

$$C_{12} = M_3 + M_5$$

$$C_{21} = M_2 + M_4$$

$$C_{22} = M_1 - M_2 + M_3 + M_6$$

Nótese que el cómputo de las matrices M_i y C_{ij} es independiente y puede hacerse en paralelo.

Asegúrese de seguir la siguiente especificación:

1. Debe escribir funciones secuenciales para ejecutar diferentes operaciones en las submatrices (adición, sustracción, multiplicación).
2. Debe agregar operaciones de CilkPlus en el código para encontrar la *mejor* forma de paralelizarlo.
3. Use el archivo provisto `strassen.cpp` como su punto de partida.
4. Genere una versión secuencial del código al compilarlo con el parámetro `-DCILK_SERIALIZE`.
5. Su programa debe correr con los siguientes parámetros:

`./strassen N`

ó

`./strassen N fileA fileB`

En el primer caso se crean matrices aleatorias de tamaño $N \times N$, mientras que en el segundo caso las matrices A y B se leen de archivos.

6. Use el siguiente comando para analizar el desempeño de su código usando diferentes números de hilos:
`CILK_NWORKERS=1 ./strassen N`

Informe

Debe crear un informe con las siguientes secciones:

1. Una estrategia general del esfuerzo de paralelización. ¿Qué patrones de programación paralela usó en la solución. ¿Qué operaciones de Cilk Plus parecieron las más apropiadas para el programa?

2. Análisis de *speedup*. Presente, para cada uno de los programas, un gráfico con el *speedup* de su solución paralela usando 2, 4, 8, 16, 32 y 64 *workers*. Este gráfico debe tener el número de *workers* en el eje *x* y el *speedup* en el eje *y*. Use el tiempo de ejecución del programa secuencial para calcular el *speedup*. Debe escoger un tamaño de problema *interesante* para su análisis.
3. Análisis de eficiencia. Presente, para cada uno de los programas, un gráfico con la eficiencia de su solución paralela usando 2, 4, 8, 16, 32 y 64 *workers*. Este gráfico debe tener el número de *workers* en el eje *x* y la eficiencia en el eje *y*. Use el tiempo de ejecución del programa secuencial para calcular la eficiencia. Debe escoger un tamaño de problema *interesante* para su análisis.

Calificación

Se evaluará la correctitud, claridad y eficiencia del código escrito, así como el apego a las directrices establecidas. La división de la evaluación es la siguiente:

- Programación algoritmo de Strassen: 70 %
- Informe con resultados: 30 %