

Asignatura: Algoritmos Distribuidos - 22629

Trabajo: 2 - 3/12/2025

Profesor: Rubén Carvajal Schiaffino

Una granja (farm) es un algoritmo distribuido donde el nodo master asigna tareas a una colección de nodos. El funcionamiento del algoritmo es el siguiente: el nodo master tiene una lista de n tareas (la misma, pero con tamaños distintos), en un primer paso se asigna una tarea a cada uno de los k nodos que realizan las tareas y luego cada vez que un nodo finaliza una tarea solicita al nodo master una nueva tarea a realizar.

La tarea a realizar es la multiplicación de dos matrices y dado que cada nodo posee varios núcleos la solución se obtendrá mediante la implementación de un algoritmo paralelo en memoria compartida.

La ejecución del programa es:

```
mpirun -np P ./t1.exe k -O data.txt
```

P: numero de nodos

k: numero de threads

O = {V: modo verboso, S: modo silencioso}

El formato del archivo de datos es:

```
n
f c1 c2
.
.
.
```

n: numero de tareas a realizar

f: numero de filas de la primera matriz

c1: numero de columnas de la primera matriz

c2: numero de columnas de la segunda matriz

La ejecución en modo verboso debe imprimir las matrices a multiplicar y su resultado, todos los elementos de la primera matriz tienen valor igual a 1.0 y todos los elementos de la segunda matriz tienen valor igual a 2.0.

La ejecución en modo silencioso debe imprimir el número de nodos participantes y el Wall-Time requerido para realizar la tarea.

Normas de Entrega

- Deadline: 8:30 del 5 enero 2026.
- El programa se codifica en lenguaje C (según el manual de normas de programación), librería pthread y librería mpich.
- Modalidad de Trabajo: Individual o Grupal (máximo dos integrantes).
- Informe: escrito en L^AT_EX (Usar plantilla de informes).
- Modalidad de envío: se sube el directorio comprimido (formato tgz, zip) que contiene todos los archivos pertinentes.
- Las eventuales copias se califican con nota 1.0 a todos los involucrados.
- El incumplimiento de la modalidad de trabajo, fecha de entrega, formato de entrega, etc. se califica con nota 1.0.

Evaluación

Se considera la programación (60%) que se evalúa del siguiente modo:

- Código fuente 75%
 - Compilación exitosa 1.5
 - Funciona con datos propuesto por el autor 3.5
 - Funciona con datos propuesto por el profesor 4.5
 - Eficiencia 5.5
 - Estilo y Documentación 7.0
- Informe 25% (se corrige solo si se cumplen los primeros tres aspectos del punto anterior)

y una interrogación individual (40%) que se realiza solo si la calificación de la programación es mayor o igual a 4.0 .

Especificación de Envío

- Código fuente, archivos de prueba, fuente del informe en L^AT_EX, informe en pdf y archivos README (con documentación adicional sobre modo de compilación, etc.) deben estar en un directorio.
- El nombre del directorio corresponde a la inicial del nombre (en mayúscula) seguido del apellido paterno con la primera letra en mayúscula y finalizará con la inicial (en mayúscula) del apellido materno. (no incluir acentos ni letra ñ o Ñ).

Ejemplo: el directorio de Juan Perez Carcamo se llamará JPerezC

En el caso que el trabajo sea realizado por dos integrantes el nombre del directorio será la concatenación de ambos nombres como se especificó en el parrafo anterior.

Ejemplo: directorio de León Cortéz Poblete y Juana Machuca Gómez: LCortezPJMachucaG

- Para crear un directorio basta escribir, por ejemplo:

```
$ mkdir JPerezC
```

y en su interior colocar los archivos requeridos.

Para comprimir el directorio a enviar se escribe, por ejemplo:

```
$ tar -czf JPerezC.tgz JPerezC
```

esto crea un archivo comprimido que se envía por correo.

Recuerde que para comprimir el directorio debe estar posicionado en el directorio padre de él.

- **NO incluir los siguientes tipos de archivos:**

Ejecutables

Archivos de respaldo. (los editores en Linux colocan el símbolo ~ al final del nombre del archivo).