Universidad Católica Andrés Bello
Escuela de Informática
Sistemas Distribuidos

Prof. Wilmer Pereira

Proyecto I (25%)

1. Objetivo

Se pretende que Uds. desarrollen una aplicación sobre el *cluster* UCAB, usando MPI, para procesamiento paralelo de archivos de texto. Se dispondrá de dos archivos:

- Un libro académico técnico, en formato texto, de cualquier área del conocimiento (al menos 500 páginas en su versión impresa).

- Una secuencia de palabras o conceptos y su definición entre comillas (al menos 4000 palabras)

El libro se debe difundir en su totalidad a cada nodo de ejecución y el archivo de secuencias de palabras (desordenadas) se repartirá por partes iguales a cada esclavo. A continuación se procesará el libro en dos fases.

En la primera fase, una vez que cada nodo de ejecución tenga el libro y su secuencia de palabras, contará todas las apariciones de cada palabra y las ordenará por orden alfabético.

En la segunda fase los esclavos harán un anillo y cada nodo substituirá la primera aparición de cada palabra por su concepto en libro y lo enviará al siguiente esclavo con todas las definiciones incluidas. Ese nodo agregará sus propias definiciones y así sucesivamente hasta llegar al último del anillo.

Para apreciar la incidencia de la cantidad de esclavos en los tiempos de ejecución, se harán dos corridas que medirán cuanto tarda en procesarse el libro por 20 y 10 esclavos separadamente. En el informe deben hacer un análisis comparativo de las dos corridas.

2. Actividades del Coordinador

Para controlar el trabajo de los esclavos se debe contar con un nodo coordinador. Su función, al comenzar, es repartir a todos el libro y separar equitativamente las palabras o conceptos para cada esclavo. Los esclavos procesarán y entregarán los datos al coordinador que consolida la información enviada por los nodos de ejecución. Esta consolidación la realiza en dos fases.

Para la primera, el coordinador recibe la lista de palabras con su número de apariciones ordenadas por orden alfabético. Debe armar un único archivo procesando inicialmente las respuestas de a pares. Es decir, al recibir la lista de palabras ordenadas que procesaron los esclavos con la cantidad de apariciones, debe crear un archivo ordenado por orden alfabético usando *merge sort*. Siempre esperará por dos respuestas para ordenar mientras los demás nodos continúan con su conteo. En consecuencia no se esperará por la finalización de todos los esclavos sino que se procesarán de a pares. Una vez que respondieron todos los esclavos, el coordinador tendrá 10 o 5 archivos temporales (dependiendo de la corrida) sobre los cuales también hará *merge sort* para ordenar en un sólo archivo las palabras que debían buscarse y contarse.

En la segunda fase simplemente recibirá el libro con todas las substituciones una vez que todos los esclavos hayan incluido sus conceptos que se le indicaron en su lista de palabras o conceptos.

3. Actividades de los Esclavo

Los nodos de ejecución son los que realizan el procesamiento que consolida el coordinador. En la primera fase, cada nodo de ejecución leerá el libro secuencialmente y buscará cada palabra de su lista contando todas las veces que aparece en el texto y substituirá la primera aparición por su definición. Enviará la lista de palabras con su cantidad de ocurrencias al coordinador (palabra número), cada par por línea. El esclavo ordenará esa lista por orden alfabético antes de enviarla.

En la segunda fase cada esclavo recibirá de su predecesor el libro con las substituciones de definiciones y agregará las suyas para posteriormente entregar al siguiente. Al final, el último del anillo entregará al coordinador.

4. Condiciones de corrida y entrega

- Deben borrar, desde el código, los archivos temporales que cada esclavo cree en los discos locales.

- Tanto el consolidado de palabras con su cantidad de ocurrencias como el libro con todas las substituciones, se guardarán en archivos en el disco del maestro

- En el informe además de los resultados del análisis de desempeño, deben responder las siguientes preguntas:

    - Los últimos *merge sort* sobre los temporales ¿Se podrían hacer más eficientemente en el coordinador con hilos? ¿Por qué?

    - ¿Tuvo trozos de código con regiones críticas? ¿Por qué?

    - ¿Es importante para este problema la sincronización de relojes? ¿Por qué?

    - Debe haber una tabla de tiempos para analizar las corridas con 20 y 10 nodos de ejecución para poder comentar los resultados.

- Se darán puntos extra si separan cada corrida equitativamente en nodos PentiumD y en nodos PentiumIV y comentan esos resultados en el análisis de desempeño.

- Para la presentación deben estar ambos miembros del grupo y se constatará el uso del *cluster* y la habilidad para manejar el gestionador de colas (`slurm`).