

Proyecto 2

Programación multihebra y mecanismos de sincronización

Integrantes: Roberto Felipe Artigues Escobar
Emilio Juan Meza Quiroz
Nicolas Eduardo Soto Soto

Profesora: Cecilia Paola Hernandez Rivas

Fecha de entrega: 24 de Noviembre de 2023
Concepción, Chile

Detalle de la Implementación

Este proyecto se ha implementado con el objetivo de analizar y comparar dos métodos distintos de sincronización en programación multihebra: el uso de semáforos y el uso de mutex combinado con variables de condición.

El programa está escrito en C++ y hace uso extensivo de las bibliotecas estándar, incluyendo `<thread>`, `<mutex>`, `<semaphore>`, y `<filesystem>`, para gestionar hilos, sincronización y operaciones de archivos.

El flujo de trabajo del programa es el siguiente:

1. **Lectura de Parámetros:** El programa comienza leyendo los parámetros de entrada que incluyen el directorio de los archivos de genomas, el valor umbral del contenido genético CG y el modo de sincronización (semaforos o mutex).
2. **Procesamiento Multihebra:** Para cada archivo en el directorio especificado, se crea un hilo que procesa el archivo de manera independiente. Esto implica calcular el contenido genético CG de todo el archivo.
3. **Sincronización y Almacenamiento en Cola:** Dependiendo del modo seleccionado, el programa utiliza semáforos o mutex con variables de condición para controlar el acceso a la cola compartida. Si el contenido genético CG de un archivo supera el umbral, su nombre se almacena en la cola.
 - En el modo de semáforos, se usa `std::counting_semaphore` para asegurar que solo un hilo a la vez pueda acceder a la cola.
 - En el modo de mutex, se emplea `std::mutex` junto con `std::condition_variable` para una gestión segura de hilos que acceden o modifican la cola.
4. **Salida:** Una vez que todos los hilos han terminado su ejecución, el programa procede a vaciar la cola compartida, imprimiendo los nombres de los archivos que cumplen con el criterio del umbral del contenido genético CG.

Evaluación Experimental

Para obtener los resultados se utilizó Linux en WSL2. Se realizaron 15 ejecuciones para ambos modos de sincronización con 10 archivos de genomas y se obtuvo su promedio.

Los comandos utilizados para ejecutar las pruebas fueron:

```
/usr/bin/time -v ./main.exe Genomas 0.6 1  
/usr/bin/time -v ./main.exe Genomas 0.6 2
```

Los resultados se muestran en la siguiente tabla:

Modo	Tiempo (s)	RAM (KB)
Semáforo	1.136	4291.2
Mutex	1.152	4342.4

Tabla 1: Comparación de tiempo y uso de memoria

Interpretación de Resultados

Los resultados obtenidos de la evaluación experimental revelan diferencias mínimas en el rendimiento entre los dos métodos de sincronización. En términos de tiempo de ejecución, el modo semáforo registró un promedio de 1.136 segundos, mientras que el modo mutex mostró un promedio ligeramente superior de 1.152 segundos. Esta pequeña diferencia sugiere que ambos métodos son casi equivalentes en eficiencia de tiempo para la tarea dada.

En cuanto al uso de memoria, el modo semáforo utilizó en promedio 4291.2 KB, frente a los 4342.4 KB del modo mutex. Aunque la diferencia es igualmente pequeña, indica una ligera ventaja en eficiencia de memoria para el modo semáforo.

En conclusión, mientras que existen diferencias leves en tiempo y uso de memoria entre los dos modos, estos son marginalmente significativos, indicando que ambos enfoques son prácticamente comparables en términos de rendimiento para el procesamiento de archivos de genomas en el contexto del experimento.