

Trabalho 1 de CLE

Grupo 9 TP2 Diogo Andrade 89265 Francisco Silveira 84802

Exercício 1 - Explicação

Foi implementada uma solução multithreading, onde cada workers é responsável por ler e processar um bloco de dados de um ou mais ficheiros. Cada bloco de dados é uma array de tamanho 1024, que contém um conjunto de palavras completas, no caso de não ser possível completar uma palavra, o ponteiro do ficheiro voltará à posição anterior à palavra.

O worker irá contar o número de palavras, como também o número de vogais tendo em conta o tamanho da palavra. O acesso aos ficheiros e a junção dos resultados dos vários workers é feito dentro de uma área partilhada de modo a não haver problemas de acesso a recursos.

Exercício 1 - Resultados

Compilar:

\$cd prog1 \$gcc main.c sharedregion.c worker.c convertchar.c -pthread

Executar:

\$./a.out <nWorkers> <pathToFiles>

Teste tempo:

Foi utilizado o dataset1 inteiro.

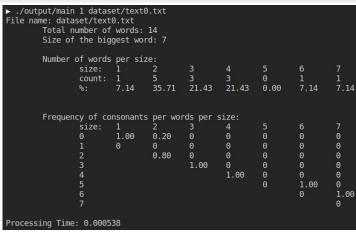


Figura 1: Exemplo de execução.

| Número de Workers | Tempo |
|-------------------|-----------|
| 1 | 0.020160s |
| 2 | 0.039441s |
| 4 | 0.042584s |

Figura 2: Teste de tempo.

Exercício 2 - Explicação

Foi implementada uma solução multithreading, onde cada workers é responsável por processar um conjunto de pares de sinais. O worker irá calcular a "circular cross-correlation" de cada par de sinais.

Após feitos todos os cálculos, quando imprimido os resultados o output irá dizer se os resultados obtidos são o esperado ou não.

O acesso aos dados dos ficheiros e a junção dos resultados dos vários workers é feito dentro de uma área partilhada de modo a não haver problemas de acesso a recursos.

Exercício 2 - Resultados

Compilar:

\$cd prog2

\$gcc main.c sharedregion.c worker.c -pthread

Executar:

\$./a.out <nWorkers> <pathToFiles>

Teste tempo:

Foi utilizado o dataset2 e o newDataset2 inteiros.

```
./output/main 1 dataset/sigVal01.bin dataset/sigVal02.bin dataset/sigVal03.bin data
set/siqVal04.bin dataset/newSiqVal01.bin dataset/newSiqVal02.bin dataset/newSiqVal03.
bin dataset/newSigVal04.bin
File name: dataset/sigVal01.bin
        Result: Expected result not found! [Expected: 0.000000 - Obtained: -3.073835]
File name: dataset/sigVal02.bin
        Result: Expected result not found! [Expected: 0.000000 - Obtained: -1.272924]
File name: dataset/sigVal03.bin
        Result: Expected result not found! [Expected: 0.000000 - Obtained: -8.019840]
File name: dataset/sigVal04.bin
        Result: Expected result not found! [Expected: 0.000000 - Obtained: 4.275348]
File name: dataset/newSigVal01.bin
        Result: Expected result founded!
File name: dataset/newSigVal02.bin
        Result: Expected result founded!
File name: dataset/newSigVal03.bin
        Result: Expected result founded!
File name: dataset/newSigVal04.bin
        Result: Expected result founded!
Processing Time: 0.768709
```

Figura 3: Exemplo de execução.

| Número de Workers | Tempo |
|-------------------|-----------|
| 1 | 0.658286s |
| 2 | 0.193755s |
| 4 | 0.061887s |

Figura 4: Teste de tempo.

Conclusão

Os resultados obtidos são o esperado. Como podemos ver no problema 1 o tempo de execução vai aumentando tendo em conta o número de workers o que faz sentido, pois o processo mais custoso é o acesso à área partilhada, onde são lidos os ficheiros. Assim os workers perdem tempo à espera da execução dos outros workers.

Já no segundo problema isso não acontece, pois a leitura dos ficheiros é feita à priori, o que ajuda nos tempos de execução. Também no problema 2 as operações feitas na área partilha são menos complexas.