Thiago Bazilio e Weberti Silva

RELATÓRIO

Trabalho apresentado Engenharia da Computação da UNIVALI - Universidade do Vale do Itajaí, para a disciplina Sistemas Operacionais.

Professor: Felipe Viel

Itajaí - Santa Catarina

Introdução

Este trabalho tem como objetivo apresentar dois projetos, cujo primeiro projeto busca mostrar uma análise comparativa de tempo de processamento utilizando bibliotecas como time.h, na qual usou-se multiplicação matricial e posicional para realizar a comparação do sistema single thread e multithread.

No segundo projeto foi feito um algoritmo de ordenação de vetores, onde o vetor é de 200 posições e foi comparado com um sistema de singlethread e multithread.

Projeto 1 códigos importantes e contexto da aplicação para a compreensão dos resultados obtidos.

Foi usado da linguagem c++ para o trabalho, na qual as funções utilizadas para obter os resultados da comparação de single thread e multithread foram:

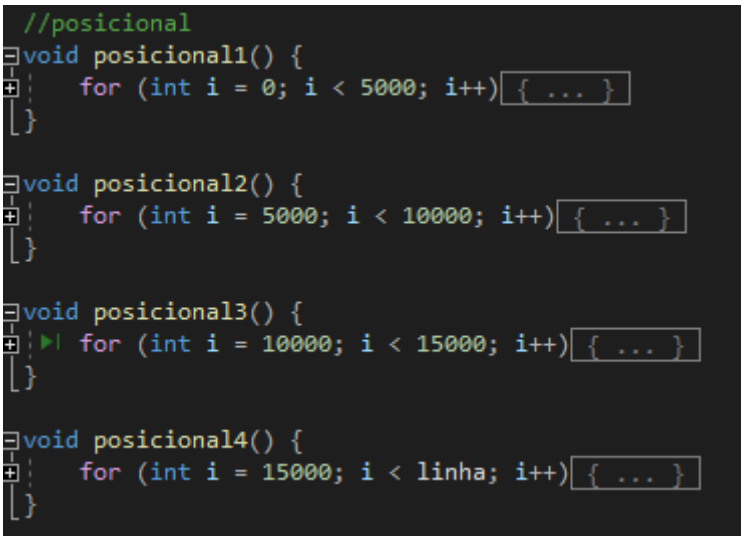
Alocar Linha: Usado para alocar as linhas das matrizes.

```
//aloca linhas
int** a = new int* [linha];

int** b = new int* [linha];

int** res = new int* [linha];
```

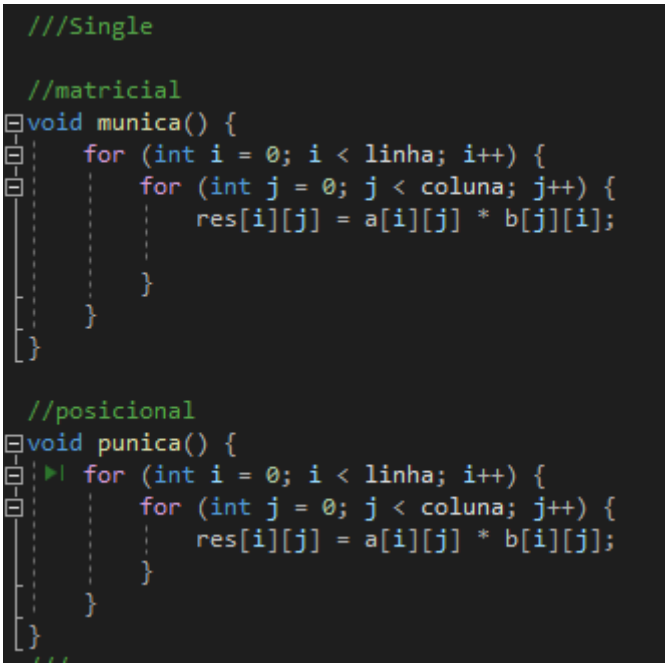
Matricial: Utilizado na multiplicação matricial das matrizes por meio de multithreads.

```
//matricial
void matricial1() {
    for (int i = 0; i < 5000; i++) {
        for (int j = 0; j < coluna; j++) { ... }
    }
}

void matricial2() {
    for (int i = 5000; i < 10000; i++) {
        for (int j = 0; j < coluna; j++) { ... }
    }
}

void matricial3() {
    for (int i = 10000; i < 15000; i++) {
        for (int j = 0; j < coluna; j++) { ... }
    }
}

void matricial4() {
    for (int i = 15000; i < linha; i++) {
        for (int j = 0; j < coluna; j++) { ... }
    }
}
```

Posicional: Utilizado na multiplicação posicional das matrizes por multithreads.



Single: Função Singlethread usado na multiplicação.



Resultados obtidos (primeiro projeto)

Simulação: Foi usada duas matrizes de 20.000 por 20.000, que foram preenchidas com números aleatórios entre 1 e 10. O processo mais demorado no projeto foi o preenchimento delas, por conta das 400 milhões de casas para preencher.

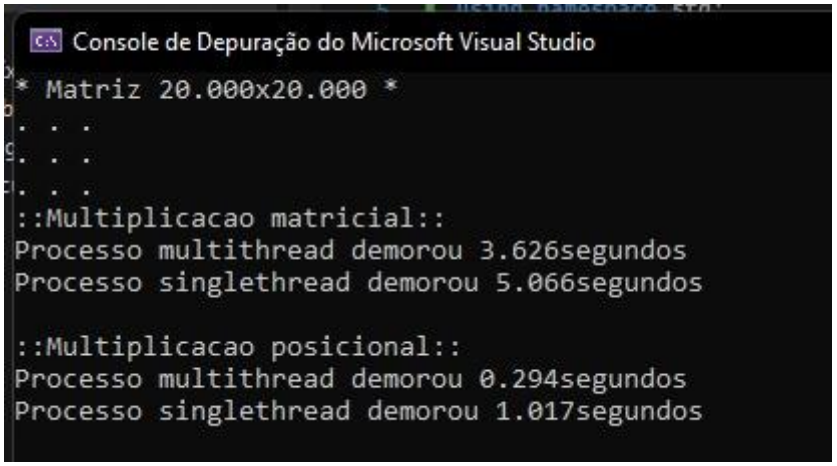


Tabela Comparativa:

Operação ::	single: (s)	multi: (s)	(%)
preencher:	54,892	6,453	850%
matricial:	5,066	3,626	140%
posicional:	1,017	0,294	345%

Conclusão

De acordo com os dados obtidos nesta análise podemos concluir que o modo singlethread se saiu pior, como já esperado. Em comparação, o multithread obteve uma satisfatória melhoria de 140% na multiplicação matricial, que é mais complexa, e de 345% na multiplicação posicional.

Projeto 2 códigos importantes e contexto da aplicação para a compreensão dos resultados obtidos.

Resultados obtidos (segundo projeto)

Simulação:

Tabela Comparativa:

Conclusão

Sendo assim, podemos analisar que a Singlethread se saiu de modo mais.... em comparação com a multithread que foi.....