



UFOP

Universidade Federal
de Ouro Preto

Trabalho Prático: Threads

**Aluno em Graduação da Universidade
Federal de Ouro Preto do curso Ciência da**

Computação:

Halliday Gauss Costa dos Santos.

Matrícula: 18.1.4093.

Área: Sistemas Operacionais.

Introdução:

O uso de threads é possível em diversas linguagens de programação. Diversos algoritmos foram inventados com o intuito de aumentar a velocidade de processamento, porém, o uso de threads, como forma alternativa, pode diminuir o tempo de execução de muitos procedimentos. No entanto, é necessário extremo cuidado com a manipulação de threads, pois geralmente atuam em uma mesma área de memória e podem alterar um dado crucial de um programa causando um mal funcionamento do mesmo. Este documento apresenta a análise da implementação de um exemplo de uso de threads em **C++** utilizando a biblioteca `pthread.h`.

Desenvolvimento:

No trabalho prático, inicialmente uma variável inteira e global 'a' foi criada e inicializada com 0. Além disso foi criado duas funções chamadas `ProcUp()` e `ProcDown()`, e ambas recebem como parâmetro o número de vezes em que serão executadas. A função `ProcUp()` incrementa em 1 o valor de 'a' e a função `ProcDown` decrementa a variável 'a' em 1. A quantidade 'n' vezes em que as funções serão executadas é igual a primeira letra do nome do aluno em ASCII concatenada com a segunda letra também em ASCII. Em seguida duas threads distintas são criadas, a `thread1` irá executar a função `ProcUp()` 'n' vezes, e a `thread2` irá executar a função `ProcDown()` 'n' vezes.

Executando o programa, o sistema irá solicitar a primeira e a segunda letra do nome do nome do usuário e fará o cálculo do número de vezes em que as funções supracitadas serão executadas. Em seguida o programa criará duas threads e atribuirão tarefas a elas. Ao executar o programa digitando os caracteres: 'H' e 'a' (relativo ao nome do aluno que desenvolveu o trabalho), a variável 'a' terá o valor 0.

Conclusão:

A variável 'a' termina com o valor zero, pois o mesmo número de vezes em que a variável 'a' é incrementada ela é decrementada, e as threads estão rodando de maneira sincronizada, caso uma dessas condições fosse falsa poderia resultar no valor de 'a' diferente de 0.

A realização desse trabalho foi de suma importância para o melhor entendimento do funcionamento de threads e como implementar essa estrutura em uma linguagem de programação.