

Laboratório 3: Manipulação de Threads em Linguagem de Programação

1. Objetivos

- Compreender as principais operações usadas em threads.
- Desenvolver aplicações usando threads.
- Explorar programação com Threads POSIX (**pthread**)

2. Materiais

- Distribuição Linux.
- Ambiente de desenvolvimento para C/C++.
- Comandos do sistema e bibliotecas de programação.

3. Descrição e Métodos

Parte 1: Análise de threads

1. Identifique no seu sistema Linux quantas threads estão em execução? Qual o processo com o maior número de threads?
2. Qual o número máximo de threads que o seu sistema suporta?
3. Verifique o tempo de execução do programa da questão 2, parte 2, considerando: 1 thread, 2 threads, 4 threads, 8 threads e 16 threads. Descreva o hardware (processador, memória e número de núcleos, tamanho da matriz usada nos testes e o tempo de execução para cada teste).

Parte 2: Programação

1. Faça um programa com **N threads** que localize um valor em um vetor de inteiros. O espaço de busca no vetor deve ser distribuído para as **N threads**.
2. Implemente um programa **multithread** com **pthread** que calcule:
 - a) a média aritmética de cada linha de uma matriz $M \times N$ e devolva o resultado em um vetor de tamanho M .
 - b) a média geométrica de cada coluna de uma matriz $M \times N$ e devolva o resultado em um vetor de tamanho N .

O programa deve gerar matrizes $M \times N$ com elementos aleatórios para arquivos; usar técnicas de paralelização de funções e de dados; ler matrizes $M \times N$ de arquivos no formato em anexo; gravar os resultados em um arquivo texto.

Importante: Fazer um *Makefile* para compilar em Linux/Unix. Use a linguagem de programação C ou C++.

As atividades da *Parte 1* devem ser entregues em um documento PDF descrevendo as respostas. As atividades da *Parte 2* devem ser entregues o código-fonte das soluções.

Compactar o documento da *Parte 1* e os arquivos da *Parte 2* em um único arquivo (tar.gz) e submeter no Moodle.