



UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS
FACULDADE DE COMPUTAÇÃO
CIÊNCIA DA COMPUTAÇÃO
SIMULAÇÃO DISCRETA

Andrey Cardoso Oliveira - 202204940036
Andréya Cristina Bentes de Paiva - 202204940005

a)

Foram avaliadas duas possíveis abordagens:

A primeira envolve a criação de um Forking Server, no qual o servidor principal, ao aceitar uma conexão, gera um novo processo para gerenciar a comunicação com o cliente específico.

A segunda opção é a implementação de um Threading Server, que, em vez de criar um novo processo, cria uma nova thread para tratar cada cliente individualmente.

b)

Para a realização dos testes de desempenho, foram criados arquivos de stress específicos tanto para o Forking Server quanto para o Threading Server. Cada arquivo de stress permite parametrizar a quantidade de clientes simulados, facilitando a avaliação do comportamento dos servidores sob diferentes cargas.

Durante os testes, o Forking Server apresentou desempenho inferior: ao realizar 1000 requisições simultâneas, além de demorar para finalizar, ocorreram erros em algumas conexões, indicando limitação no gerenciamento de múltiplos processos.

Por outro lado, o Threading Server demonstrou desempenho superior. Com 1000 requisições, todas foram processadas corretamente e de forma rápida

Esses resultados mostram que, para este projeto, a abordagem baseada em threads foi mais eficiente e escalável do que a abordagem baseada em processos (fork), especialmente em cenários com grande volume de conexões simultâneas.

c)

O compilador foi implementado utilizando um servidor Threading, devido ao seu melhor desempenho em testes de estresse, enquanto a interface foi desenvolvida em Python.

Repositório: <https://github.com/andreydedey/Lab-distributed-systems>