Alunos: Autierus Pereira, Rafael Wurzius e Vanessa França

Relatório Sistema de Carros de Aplicativo com Threads

Para o desenvolvimento de um sistema de carro por aplicativo que processa múltiplas requisições simultâneas, a arquitetura proposta utiliza threads para gerenciar as requisições dos clientes, garantindo a atribuição eficiente dos carros disponíveis e o tratamento de requisições em fila de espera. O objetivo é validar a abordagem multithread, testar a robustez do sistema em um cenário de alta concorrência e demonstrar a capacidade de enfileirar e processar solicitações.

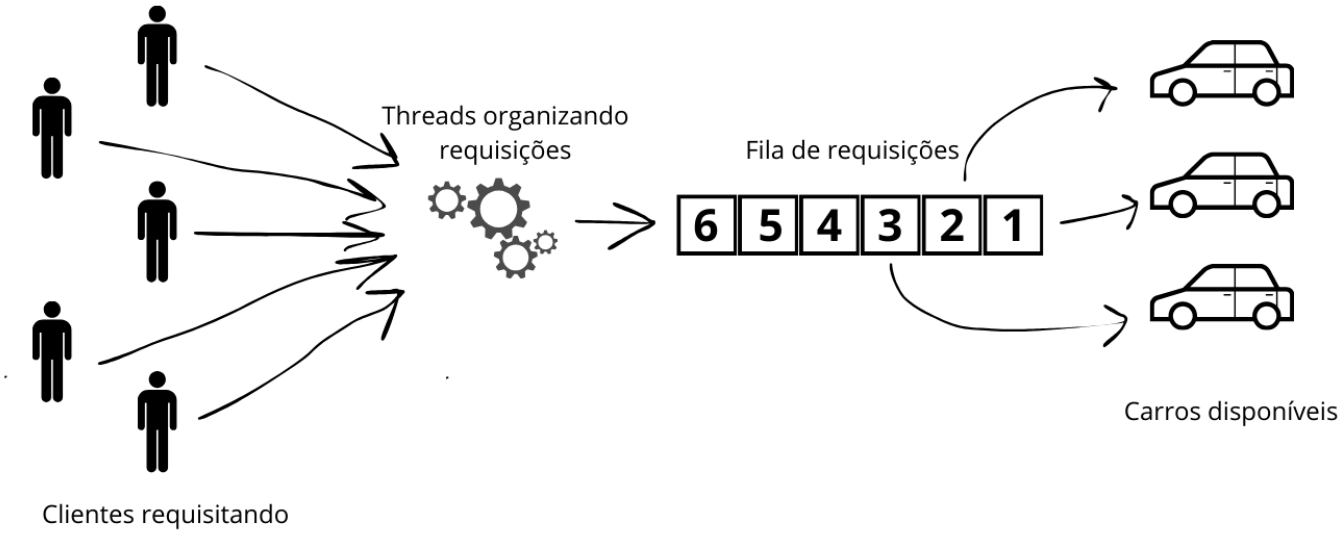
Nesse cenário, o “produtor” é responsável por simular os clientes enviando requisições de corrida ao servidor. Cada requisição é enviada em uma thread separada, simulando um cenário real onde múltiplos clientes solicitam corridas simultaneamente. Já o “Consumidor” representa o servidor que gerencia as requisições de corrida. Ele utiliza uma fila para armazenar as requisições que aguardam por um carro disponível. Uma única thread é responsável por processar essas requisições, atribuindo carros conforme eles se tornam disponíveis e enviando confirmações de serviço para os clientes.

O consumidor (servidor) escuta requisições de corrida de múltiplos clientes simultaneamente. Cada conexão é tratada em uma thread dedicada, que recebe a requisição e a coloca em uma fila de espera e uma thread específica no servidor processa a fila de requisições. Se houver carros disponíveis, a thread atribui um carro à requisição, remove a requisição da fila e envia uma confirmação ao cliente.

Dessa forma, são evitados conflitos, tais como a atribuição de um mesmo carro para dois clientes ou vice-versa, o que ocorre, por exemplo, quando uma ação de alteração de “status” do carro é realizada por uma thread, porém outra realiza a mesma operação simultaneamente, antes que o status seja de fato alterado.

Por fim, se não houverem carros disponíveis no momento da requisição, a mesma permanece na fila até que um carro se torne disponível. As requisições são processadas em ordem de chegada. Após a atribuição de um carro, o consumidor envia uma confirmação ao produtor, informando que o carro foi designado com sucesso.

O diagrama a seguir representa fluxo de atividades realizadas pelo sistema:



Link para o repositório Github: https://github.com/RafaelWurzius/ProjetoProgramacaoParalelaMulticore.git