

UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO – UFRSA

Centro de Ciências Exatas e Naturais – CCEN

Departamento de Computação - DC

Graduação em Ciência da Computação

Disciplina: Estrutura de Dados II

Prof.: Paulo Henrique Lopes Silva

Prática Offline 2

1. Cache Eviction.

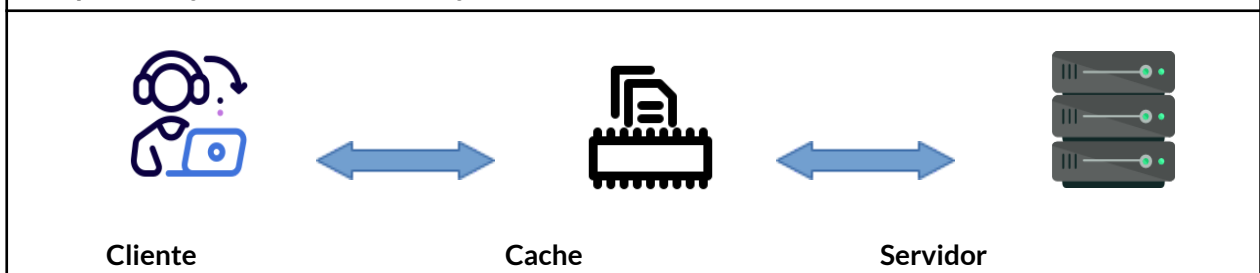
- **Cache eviction** é o processo de remoção de itens do cache quando ele atinge sua capacidade máxima, a fim de liberar espaço para novos dados.
- Esse processo é essencial para manter a eficiência do cache, evitando que ele fique cheio e incapaz de armazenar novos dados.

2. Considere a necessidade do desenvolvimento de uma base de dados sobre ordens de serviço (OS) para reparar falhas em equipamentos.

- Tarefa:
 - **Simulação de uma aplicação cliente/servidor, onde:**
 - Uma classe ou conjunto de classes implementa o cliente que pode fazer buscas na base de dados;
 - Uma classe ou conjunto de classes implementa o servidor que controla a base de dados.
 - **Sobre as entidades:**
 - Cada ordem de serviço possui dados como código, nome, descrição e hora da solicitação.
 - **Sobre a cache e a base de dados:**
 - A base de dados deve ser implementada como uma Tabela Hash.
 - A modelagem da tabela hash (função de dispersão e tratamento de colisões) faz parte da interpretação do projeto (use orientação a objetos). Justifique as escolhas.
 - É importante que a base de dados seja uma tabela hash redimensionável.
 - A cache pode ser implementada como uma tabela hash para 20 elementos e deve utilizar:
 - A cache só deve ser preenchida a partir de buscas.
 - Crie uma função para fazer um número de buscas que preencha a cache.
 - Operação de busca na cache.
 - Buscas devem ser feitas primeiro na cache. Caso o item de busca não esteja na cache, a busca é redirecionada para a base de dados.
 - Alterações devem ser feitas na base de dados e refletidas na cache.
 - Política **Random** para inserções e remoções na cache.
 - Ou alguma outra política onde a busca seja rápida. Justificar.
 - Justifique a escolha da função de dispersão e, eventualmente, do mecanismo de tratamento de colisões para as tabelas da base de dados e da cache.

- **Requisitos da simulação:**
 - Tanto o cliente como o servidor vão ser executados no mesmo computador (simulados).
 - Entre o cliente e o servidor existe a cache.
 - Clientes podem fazer consultas (**buscas**) usando o código.
 - Outras operações que um cliente pode fazer são:
 - **Cadastrar OS.**
 - **Listar** ordens de serviço (todos os dados de todas as ordens de serviço).
 - **Alterar OS.**
 - **Remover OS.**
 - **Acessar a quantidade** de registros.
 - Servidor atende às requisições do cliente.
 - O servidor deve manter uma espécie de log informando o estado da tabela hash a cada operação (sugiro inserir essas informações em um arquivo).
 - Além disso, deve mostrar os itens da cache a cada operação.
 - Para mostrar a política de **cache eviction** funcionando.
 - Um cliente pode fazer a autenticação para acessar a aplicação?
 - É facultativo.
 - Tratamento de erros e exceções onde for possível (interpretativo).
- **Execução da simulação:**
 - Inicia o programa:
 - 70 ordens de serviço devem ser adicionadas na base de dados do servidor.
 - Após isso, mostrar a interface do cliente.
 - Realizar:
 - Três consultas.
 - Uma listagem.
 - Dois cadastros seguidos de listagens.
 - Duas alterações seguidas de listagens.
 - Duas remoções seguidas de listagens.
 - Escolha operações de forma a evidenciar o funcionamento do **cache eviction**.

3. Representação Gráfica da Simulação.



4. Avaliação:

- O trabalho vale 60% da nota da 2ª unidade.
- Deve ser desenvolvido na linguagem Java usando orientação a objetos.
 - Data de entrega: 25/09/2024, somente via sigaa.
 - Formato de envio: seu-nome-pratica-off-2.zip.
- **O projeto é individual.**
 - Em caso de alguma dúvida, o(a) aluno(a) poderá ser chamado para esclarecimentos.
- **Em caso de comprovação de cópia ou fraude, os trabalhos envolvidos receberão a nota ZERO.**
- Entregou fora do prazo?
 - O desconto na nota será de 20%.
 - Data limite: 26/09/2024.

- Entregas não serão aceitas após a data limite.
- Não há reposição de trabalhos práticos.