UNIFOR
UNIVERSIDADE DE FORTALEZA
FUNDAÇÃO EDSON QUEIROZ
ENSINANDO E APRENDENDO

ASSINALE A VERIFICAÇÃO CORRESPONDENTE A ESSE TRABALHO: AV 1 ☐ AV 2 ☐ AV 3 ☐

CENTRO:_____ DISCIPLINA:_____ DATA: ___/___/___ PROF(ª):_____

ALUNO(A): _____ MATRÍCULA: _____

ALUNO(A): _____ MATRÍCULA: _____

ALUNO(A): _____ MATRÍCULA: _____

INSTRUÇÕES PARA O DESENVOLVIMENTO DO TRABALHO

- Trabalho é em Equipe (até 4 pessoas)
- O Trabalho vale 30% da nota da disciplina.
- O Trabalho será apresentado pela equipe. Será marcado um dia para a equipe mostrar a implementação e a execução da aplicação.

Implementar um índice hash estático

1. Interface gráfica obrigatória ilustrando as estruturas de dados e o funcionamento de um índice hash estático.

2. Funcionalidades principais:

a. Construção do índice;
b. Busca por uma tupla a partir da entrada de uma chave de busca usando o índice construído;
c. Fazer um table scan dos X primeiras tuplas.

3**. Entidades/estruturas a serem implementadas (usando POO como padrão):**

a. **Tupla:** representa uma linha da tabela, contém o valor da chave de busca e os dados da linha.
b. **Tabela:** contém todas as tuplas construídas a partir do carregamento do arquivo de dados.
c. **Página**: estrutura de dados que representa a divisão e alocação física da tabela na mídia de armazenamento.
d. **Bucket**: estrutura de dados que mapeia chaves de busca em endereços de páginas.
e. **Função hash**: mapeia uma chave de busca em um endereço de bucket. Deve ser escolhida/projetada pela equipe.

4. Parâmetros:

a. **Arquivos de dados:** contém os dados que serão usados para popular uma ou mais tabelas. Para este trabalho será usado um arquivo com 466 mil palavras do idioma Inglês, disponível em: https://github.com/dwyl/english-words. Este arquivo txt tem somente uma palavra por linha. Esta única palavra é única no arquivo, podendo ser considerada chave.
b. **Tamanho da página**: entrada de usuário que determina o tamanho de cada página individualmente.
c. **Quantidade de páginas**: número máximo de páginas usadas [1] para dividir o conteúdo da tabela.
d. **Número de buckets (NB):** calculado, onde NB > NR / FR. NR é a cardinalidade da tabela (número de tuplas) e FR é o número de tuplas endereçadas por bucket.

[1] Se o usuário escolheu o tamanho da página então a quantidade de páginas é um parâmetro calculado. Os dois parâmetros são mutuamente exclusivos como entrada.

e. **Tamanho dos buckets (FR):** número máximo de tuplas endereçadas por bucket, depende da função hash implementada.

f. **Chave de busca de uma tupla específica:** depois que o índice é construído o usuário pode entrar com uma chave de busca para que o sistema retorne a tupla associada. Deve existir um local na interface gráfica para ser digitada a chave que será buscada. O registro retornado e em qual página está localizado devem ser mostrados na interface gráfica

g. Quantidade de registros do Table Scan: entradaq de usuário para fornecer qual a quantidade de registros que deve ser mostrada a partir do início. Deve existir um local na interface gráfica para ser digitado este número. Os registros devem ser mostrados em na interface gráfica.

5. Problemas:

a. A implementação do índice deve levar em consideração as colisões, ou seja, deve ser implementado um algoritmo de resolução de colisões.

b. A implementação do índice deve levar em consideração o transbordamento dos buckets (bucket overflow), ou seja, deve ser implementado um algoritmo de resolução de overflow.

6. Estatísticas:

a. Deve ser calculada e mostrada a taxa de colisões.

b. Deve ser calculada e mostrada a taxa de overflows.

c. Deve ser calculado e mostrado uma estimativa de custo (acessos a disco), quando uma chave de busca é entrada (funcionalidade b).

7. Funcionamento em passos:

a. O arquivo de dados é carregado em memória.

b. Cada linha do arquivo deve gerar uma tupla, que será adicionada à tabela.

c. As tuplas da tabela devem ser divididas em páginas, de acordo com o tamanho das páginas.

d. NB buckets de tamanho FR são criados.

e. A função hash é aplicada à chave de busca de cada tupla; a chave de busca e o endereço da página onde a tupla foi armazenada são adicionadas ao bucket cujo endereço foi calculado pela função hash.

Critério	Notas
1. Interface gráfica	1,0
2. Carga de Dados nas páginas	1,5
3. Entrada para Tamanho da página	1,0
4. Cálculo da quantidade de páginas	1,0
5. função hash	1,0
6. Cálculo da quantidade de buckets	0,5
7. Funcionamento com pesquisa	2,0
8. Deve ser calculada e mostrada a taxa de colisões.	0,5
9. Deve ser calculada e mostrada a taxa de overflows.	0,5
10. Table Scan	0,5
11. Deve ser calculado e mostrado uma estimativa de custo (acessos a disco)	0,5