

## Lista de Exercícios sobre Hash

Consultar os arquivos:

Aula7.pdf, Aula8.pdf, Aula9.pdf

- 1) Qual a idéia do uso de uma tabela Hash?
- 2) Qual a diferença entre tabela de hashing e função de hashing?
- 3) Cite duas desvantagens da tabela hash em relação à árvore binária de busca.
- 4) Como funciona o hashing fechado (endereçamento aberto) e o hashing aberto (encadeado)?
- 5) Como deve se comportar a função de hash na pesquisa e na inserção feitas na tabela de hash?
- 6) Quando ocorre uma colisão?
- 7) Ao implementar uma tabela de hash, o cálculo do endereço gera dois problemas. Explique.
- 8) Implemente o que se pede:

### Hashing Aberto ou Encadeado: Exercício

Com base no que foi apresentado defina a(s) estrutura(s) de dados necessária(s) para a implementação de uma tabela hashing aberta.

```
#define numEntradas 8
typedef struct _Hash
{
    int chave;
    struct _Hash *prox;
} Hash;
typedef Hash* Tabela[numEntradas];
```



Implemente a função de inserção de uma chave na tabela de hashing aberta em questão.

```
void inserirHash(Tabela tabela, int n){...} //n é o elemento a ser inserido.
```

A função `inserirHash` usa a função de hashing abaixo que também deve ser implementada:

```
int funcaoHashing(int num){...} // dado um número num, retorna a posição dele usando o método da divisão.
```

Implemente a função que retorna um ponteiro para o elemento da tabela de hash(o elemento está presente na tabela)

```
Hash* localizarHash(Tabela tabela, int num){...} //num o elemento sendo procurado.
```

Implemente a função de remoção de uma chave da tabela de hash em questão:

```
void excluirHash(Tabela tabela, int num){...} // num a chave a ser excluída
```

9) Implemente o que se pede:

## Tabelas de hash fechada: Exercício

Com base no que foi apresentado defina a(s) estrutura(s) de dados necessária(s) para a implementação de uma tabela hashing fechada com tentativa linear.

```
#define tam 8
typedef struct
{
    int chave;
    char livre; /* L = livre, O = ocupado, R =
removido*/
}Hash;
typedef Hash Tabela[tam];
```

201



Agora, implemente a função de inserção de uma chave na tabela hashing fechada em questão.

```
void inserir(Tabela tabela, int n) {
```

```
int funcaoHashing(int num)
{
```

Implemente a função de remoção de uma chave na tabela hashing fechada em questão.

```
void remover(Tabela tabela, int n)
{
```

```
int buscar(Tabela tabela, int n)
{
```

10)

11)