

NOME:

Disciplina: Organização e Arquivos de Dados

PROVA BIMESTRAL (VALOR 10,0)

1. **Faça um programa que apresente o menu abaixo. As operações devem ser implementadas em uma tabela hashing. Utilize o tipo do produto como chave da tela. Suponha que existam apenas os tipos informados abaixo (em número de 4) e que os dados de cada entrada da tabela sejam armanezados na forma de lista encadeada.**

Menu

- 1- Inserir produto
- 2- Consultar todos os produtos cadastrados de um tipo
- 3- Contar quantos produtos estão cadastrados em cada tipo
- 4- Sair

Obs.

Na opção 1, deve ser inserido um produto de cada vez, sendo que para cada um deles o usuário deve fornecer a descrição e o tipo (A-Alimentação, H-Higiene, L-Limpeza, V-Vestuário)

Na opção 2, o usuário deve digitar a letra que corresponde ao tipo a ser consultado, e todos os produtos do tipo fornecido devem ser listados(listar apenas a descrição). Caso não tenha nenhum produto do tipo fornecido, mostrar mensagem.

Na opção 3, o programa deve mostrar quantos produtos estão cadastrados em cada tipo.

2. **Faça um programa que apresente o menu abaixo. As operações devem ser implementadas em uma tabela hashing. Utilize o mês de aniversário como chave. Suponha que a tabela possua o tamanho 20 e que seja implementada como uma de endereçamento aberto (tentativa linear).**

Menu

- 1 – Cadastrar amigo (nome, dia e mês de aniversário)
- 2 – Consultar aniversariantes de um mês
- 3 – Contar as pessoas com idade superior a 18
- 4 – Excluir uma pessoa
- 5 – Sair

3. **Uma árvore binária completa pode ser armazenada de maneira eficiente em um vetor (os nós no vetor seguem a mesma ordem de um percurso em largura na árvore).**

- Escreva uma função que verifica se a árvore armazenada em um vetor é um heap de máximo ou mínimo, isto é, se todo nó possui uma chave maior ou igual às chaves dos seus filhos.

Menu

Heap-Max/Inserir

Heap-Min/Inserir

Heap-Max/Remover

Heap-Min/Remover

Heap

Consultar toda a lista

Sair