

Aula 09 – Exercícios

Parte 1 – Escrita (individual – entregue via Formulário no Google Classroom)

Exercício 1: em seu caderno, mostre a inserção das chaves 30, 51, 15, 44, 48, 22, 18, 5, 7, 41, 37 em uma tabela hash de 15 posições, usando linear probing. A função hashing a ser empregada é o método da divisão. Não, não pode fazer no computador. É à mão!

Parte 2 – Codificação (individual – entregue via Run Codes)

Exercício 2: escreva, em linguagem C pura (sem nenhum comando próprio de C++), um programa que leia um valor N e aloque um vetor de inteiros de N posições para implementação de uma tabela hash usando linear probing. A função hashing a ser empregada é o método da divisão. A seguir, leia outro número inteiro representando uma opção do usuário. As ações a serem executadas dependerão da opção lida, sendo:

- Se a opção lida for **1**, seu programa deve ler um novo valor inteiro e inserir na tabela hash.
 - Se a opção lida for **2**, seu programa deve ler um novo valor inteiro e pesquisá-lo na tabela hash. Se o valor pesquisado estiver presente na tabela, imprima o seu índice no vetor. Se o valor pesquisado não estiver presente na tabela, imprima o valor -1 como resposta. Pule uma linha após imprimir sua resposta.
 - Se a opção lida for **3**, seu programa deve ler o valor de um índice do vetor e apagar aquela posição.
 - Se a opção lida for **4**, seu programa deve imprimir, em uma única linha, todos os elementos da tabela hash, separados entre si por um único espaço. Pule uma linha após realizar a impressão da resposta.
 - Se a opção lida for **9**, seu programa deve finalizar. Note que seu programa encerra apenas nesta opção. Ao terminar de processar qualquer outra opção, o programa deve retornar ao loop inicial para ler a próxima ação que o usuário deseja executar.
-

Exercício 3: escreva, em linguagem C pura (sem nenhum comando próprio de C++), um programa que implemente uma tabela hash de 13 posições usando open hashing. A função hashing a ser empregada é o método da divisão e a disposição dos elementos nas listas segue a ordem de inserção. A seguir, leia um número inteiro representando uma opção do usuário. As ações a serem executadas dependerão da opção lida, sendo:

- Se a opção lida for **1**, seu programa deve ler um novo valor inteiro e inserir na tabela hash.
 - Se a opção lida for **2**, seu programa deve ler um valor inteiro e pesquisá-lo na tabela hash. Se o valor pesquisado estiver presente na tabela, imprima-o. Se o valor pesquisado não estiver presente na tabela, imprima um “x” minúsculo como resposta. Pule uma linha após imprimir sua resposta.
 - Se a opção lida for **3**, seu programa deve imprimir toda a tabela hash, no seguinte formato:
[0]: A B C D
[1]: F G
[2]: H I J K L
etc...
Onde, entre colchetes, está o índice do vetor e, na sequência, os elementos inseridos naquela respectiva lista. Os elementos de uma lista são separados entre si por um único espaço. As listas são impressas uma em cada linha.
 - Se a opção lida for **9**, seu programa deve finalizar. Note que seu programa encerra apenas nesta opção. Ao terminar de processar qualquer outra opção, o programa deve retornar ao loop inicial para ler a próxima ação que o usuário deseja executar.
-

Observação: este exercício está dividido em uma parte escrita e uma parte prática de codificação. A entrega só será válida se as duas partes forem entregues.