UNIVERSIDADE ESTADUAL DE CAMPINAS
Instituto de Computação

Disciplina	Entrega
MC202	**10/07/2020, 23:59**

Professor
Iago Augusto de Carvalho
Monitores
Arthur (PAD), Brenner (PED), Deyvison (PED), Enoque (PED), Matteus (PED), Thiago (PAD).

Atividade de Laboratório 10

1 Introdução

Denilson Noveira (*DN*) começou seus estudos sobre tabela hash e ficou fascinado. Depois de conhecer todas as suas possíveis aplicações, *DN* decidiu criar seu próprio modelo de tabela hash que o mesmo resolveu chamar de *DhashN*. O método *DhashN* consiste em aplicar uma função de hash $\mathbf{h(x)} = x$ % tamanho_da_tabela e armazenar o elemento na posição resultante da função e, em caso de colisão, adiciona o elemento na posição livre posterior mais próxima (considere que a tabela **hash** é **circular**, então a próxima posição da ultima posição é a primeira).

A Figura 1 mostra um exemplo de como funciona esta tabela hash. Neste exemplo, temos que o tamanho da tabela é de 10 posições. Queremos inserir, na sequência, os números 28, 55, 71, 67, 11, 10, 90, 44. A inserção dos números é realizada como a seguir. Notem que existiram 4 conflitos na inserção destes números, sendo uma colisão na inserção do número 11 e três colisões na inserção do número 90.

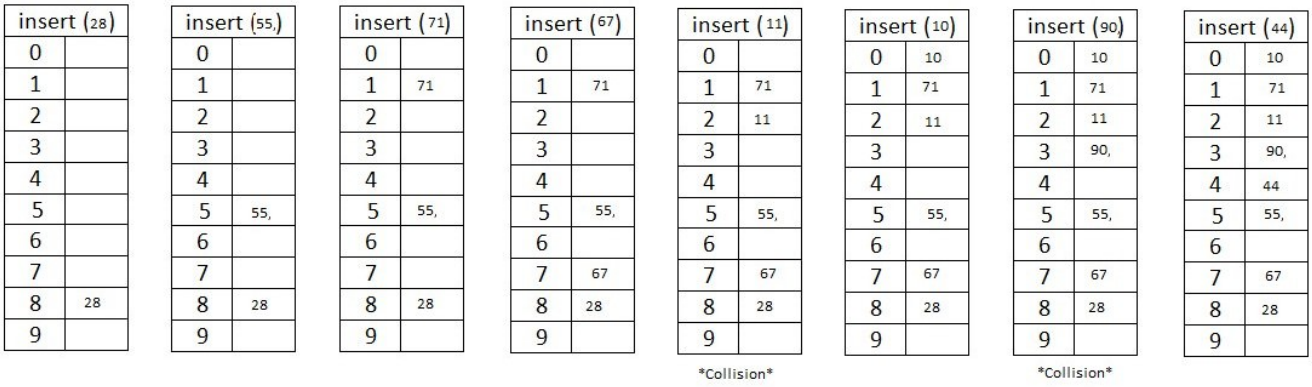

Figura 1: Exemplo de inserção dos números 28, 55, 71, 67, 11, 10, 90, 44 em uma tabela hash de tamanho 10

- Fazemos 28 % 10 = 8. Então, inserimos o número 28 na posição 8.
- Fazemos 55 % 10 = 5. Então, inserimos o número 55 na posição 5.
- Fazemos 71 % 10 = 1. Então, inserimos o número 71 na posição 1.
- Fazemos 67 % 10 = 7. Então, inserimos o número 67 na posição 7.
- Fazemos 11 % 10 = 1. Então, tentamos inserir o número 11 na posição 1. Entretanto, já existe um número lá (o número 71), o que configura uma colisão. Então, tentamos inserir o número 11 na próxima posição. Verificamos que a próxima posição (posição 2) está livre livre. Assim, o número 11 é colocado na posição 2.
- Fazemos 10 % 10 = 0. Então, inserimos o número 10 na posição 0.

- Fazemos 90 % 10 = 0. Então, tentamos inserir o número 90 na posição 0. Entretanto, já existe um número lá (o número 10), o que configura uma colisão. Então, tentamos inserir o número 90 na próxima posição. Verificamos que a próxima posição (posição 1) também está ocupada, o que configura uma nova colisão. O mesmo acontece com a posição 2. Por fim, o número 90 é colocado na posição 3. Notem que, neste caso, existiram 3 conflitos.
- Fazemos 44 % 10 = 4. Então, inserimos o número 44 na posição 4.

Porém, *DN* sabe que se uma tabela hash possuir muitas colisões, ela acaba sendo um pouco ineficiente. Como ele não é muito bom em programação, ele pediu sua ajuda para contar o número de colisões que o método *DhashN* causaria para algumas sequências de números. Para seus experimentos, *DN* deseja manter a mesma função de hash, isto é, $\mathbf{h(x)} = x$ % tamanho_da_tabela.

2 O que deve ser feito

2.1 Algoritmos

Deve-se implementar um algoritmo que dado um *N*, indicando a quantidade de elementos que serão digitados e o tamanho da tabela hash, deve retornar o número de colisões utilizando o método *DhashN*.

2.2 Restrições

1. O código deve ser **feito em C**;
2. Você deverá implementar qualquer estrutura de dados que utilizar;

2.3 Entrada

A primeira linha da entrada é composta por um inteiro *N*, onde $N < 32767$. Este número representa tanto o tamanho da tabela quanto o número de elementos que deverão ser inseridos. A segunda linha consiste em *N* inteiros, representando os números utilizados para o teste de *DN*.

2.4 Saída

A saída de seu programa deve ser um inteiro, **seguido** de uma quebra de linha, indicando qual o número de colisões que ocorre na tabela hash utilizando o método de *DN*.

2.5 Exemplos de Entrada e Saída

Entrada	Saída
5 0 1 2 3 4	0
5 0 1 1 2 3	3
5 4 9 14 19 5	9

3 Entrega

Você deve entregar seu código pelo **Susy**, através do link https://susy.ic.unicamp.br:9999/mc202defg/, contendo um único arquivo **main** nomeado de *lab10.c* e até 4 outros arquivos *.c* e *.h* (**podendo até ser nenhum outro**).

4 Nota

Essa atividade de laboratório possui peso 4.

5 Dúvidas

Em caso de dúvidas, entre em contato com um dos monitores ou o professor da disciplina a qualquer momento.