

Introdução aos Algoritmos e Estruturas de Dados

(<https://fenix.tecnico.ulisboa.pt/disciplinas/IAED7645111326/2016-2017/2-semester>)

Exercícios

1*) Considere a seguinte função de dispersão:

```
int hash(int value, int M) {  
    return value % M;  
}
```

Usando uma tabela de dispersão por encadeamento externo (external chaining) para guardar elementos com as seguintes chaves

0, 32, 1, 35, 2, 33, 38, 10, 4, 3 e 6,

e a função de dispersão definida em cima, e sabendo que $M = 10$, qual ou quais são as chaves dos elementos guardados na posição 3 da tabela (A primeira posição da tabela é a posição zero)?

2) Qual o número total de conflitos (elementos adicionados a uma posição já contendo pelo menos um elemento) quando o último valor da sequência **< 17, 7, 28, 12, 0, 25, 37, 11 >** é introduzido numa tabela de dispersão de dimensão 10 com resolução por encadeamento externo (external chaining), inicialmente vazia, sabendo que a função de hash é $\text{hash}(k) = k \bmod 3$.

3*) Qual a posição em que é colocado o último valor da sequência **< 17, 7, 28, 12, 0, 25, 37, 11, 24 >** ao serem introduzidos numa tabela de dispersão de dimensão **$M=13$** por linear probing, inicialmente vazia, sabendo que a função de hash é $\text{hash}(k) = k \bmod M$?

4) Considere uma tabela de dispersão com resolução por procura linear (linear probing), que permite guardar números inteiros. A tabela tem dimensão **$M = 10$** , e a respectiva função de dispersão é $\text{hash}(k) = k \bmod M$. Indique, para a inserção na tabela da sequência **< 10, 18, 5, 25, 46, 101, 39, 17 >**, qual será o índice da entrada da tabela em que é inserido o último elemento ?

5*) Considere uma tabela de dispersão com resolução por dispersão dupla (double hashing), com dimensão **$M = 10$** , em que as funções de dispersão são dadas por:

```
hashone(k) = k mod M  
hashtwo(k) = (1 + 3k)
```

Qual o índice da posição na tabela em que é colocado o último valor da sequência **< 10, 12, 7, 9, 3, 11, 2 >**, assumindo que a tabela se encontra inicialmente vazia ?

6) Qual a sequência de inserção numa árvore de pesquisa binária (binary search tree) inicialmente vazia que resulta numa árvore equilibrada ?

- a. < 23, 19, 21, 15, 18, 16, 12, 25 >
- b. < 23, 25, 19, 15, 21, 18, 16, 12 >
- c. < 12, 15, 16, 18, 19, 21, 23, 25 >
- d. < 25, 23, 21, 19, 18, 16, 15, 12 >
- e. < 18, 21, 23, 25, 15, 19, 16, 12 >
- f. < 18, 23, 16, 21, 15, 19, 12, 25 >
- g. < 23, 18, 21, 16, 15, 12, 19, 25 >

7*) Qual a operação de rotação que transforma a árvore de pesquisa binária, resultante da inserção da sequência abaixo, numa árvore equilibrada ?

< 15, 12, 17, 21, 23 >

- a. rotR(15)
- b. rotL(15)
- c. rotR(17)
- d. rotL(17)
- e. rotR(21)
- f. rotL(21)
- g. rotR(23)

Nota: considere que rotL e rotR são as operações de rotação para a esquerda e para a direita, respectivamente.

8*) Considere uma árvore AVL para inteiros inicialmente vazia onde são inseridos sequencialmente os elementos do seguinte vector: **< 10, 8, 9, 7, 16, 3, 50, 15, 6, 11 >**. Desenhe a árvore resultante e indique a sequência de elementos visitados por uma travessia post-order.

9*) Considere a árvore resultante do exercício anterior e elimine o elemento 16. Desenhe a árvore resultante e indique a sequência de elementos visitados por uma travessia pre-order.

Soluções

Soluções: 1) 3 e 33; 2) 5; 3) 5; 4) 2; 5) 6; 6) e; 7) d; 8) <6, 3, 8, 7, 10, 15, 11, 50, 16, 9>; 9) <9, 7, 3, 6, 8, 15, 11, 10, 50> ou <9, 7, 3, 6, 8, 11, 10, 50, 15>