

Lista de Exercícios (2)
Estruturas de Dados, CIC, UnB
Prof. Díbio

1. Tabelas de dispersão, ou espalhamento (*i.e. hashing*), são apropriadas para realizar buscas rápidas em um conjunto de informações. Através do uso de uma chave, uma função de espalhamento irá mapear a chave ao índice do elemento procurado. Descreva, usando o método da divisão, o seguinte:

2. Considerando 10 posições disponíveis, como seriam mapeadas as chaves 123460, 123786, 123777, 134595, 156188, 156182, 165909, 187324, 198723, 124321? Como?

0	1	2	3	4	5	6	7	8	9

3. Considere agora que as chaves não sejam inteiros, como mapear “ROBINHO”, “NEYMAR”, “JULIANA PAES”, “CLEO PIRES”, “RENATO RUSSO”?

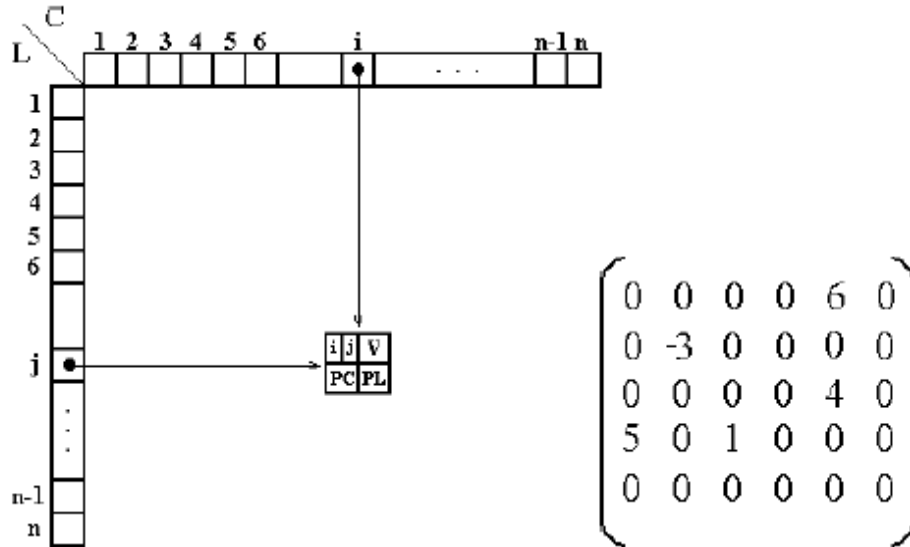
0	1	2	3	4	5	6	7	8	9

4. Caso o método para gerar chaves inteiras a partir de sequências de caracteres gere a mesma chave para mais de uma sequência teremos colisões. Duas maneiras para lidar com colisões são endereçamento aberto, e encadeamento, descreva os dois métodos e como eles seriam aplicados às seguintes sequências: “ROBINHO”, “NEYMAR”, “JULIANA PAES”, “CLEO PIRES”, “RENATO RUSSO”, “DADA MARAVILHA”, “OSCARITO”, “JK”, “SABRINA”, “JOAO”, “REGINA”.

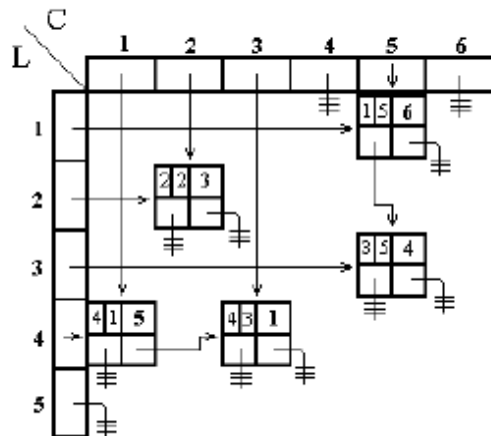
5. Escreva, em linguagem C, todas as funções necessárias para TADs dos dois métodos de colisão da questão 4.

6. Desenvolva um mecanismo para detectar quando todas as possíveis posições de re-espalhamento (“*rehash*”) de determinada chave foram pesquisadas.

7. Uma matriz esparsa é aquela onde grande parte dos seus elementos tem valor zero. Ela pode ser representada graficamente como segue:



para a matriz do lado direito a representação seria:



L e C são vetores que apontam para os nós. Sabendo disso, faça:

- Defina a Estrutura de dados para a Matriz Esparsa;
- Escreva a função `CriaMatrizEsparsa(...)`
- Escreva a função `SomaElemento(K, Matriz)` onde k é o valor a ser somado e `Matriz` é a matriz esparsa. Observe que nessa operação, nós podem aparecer e outros serem removidos...

8. Uma Árvore é uma estrutura de dados apropriada para organizar dados hierárquicos, descreva e dê exemplos das seguintes estruturas:

- Árvore Genérica
- Árvore Binária
- Árvore Binária de Busca

9. Descreva todas as operações fundamentais de uma Árvore Binária de Busca, e implemente funções para o TAD considerando uma lista encadeada como primitiva.

10. Escreva uma função que decida se uma dada árvore binária é ou não é de busca.

11. Desenhe uma Árvore Binária de Busca resultante da inserção em uma árvore vazia, dos seguintes itens, nesta ordem:

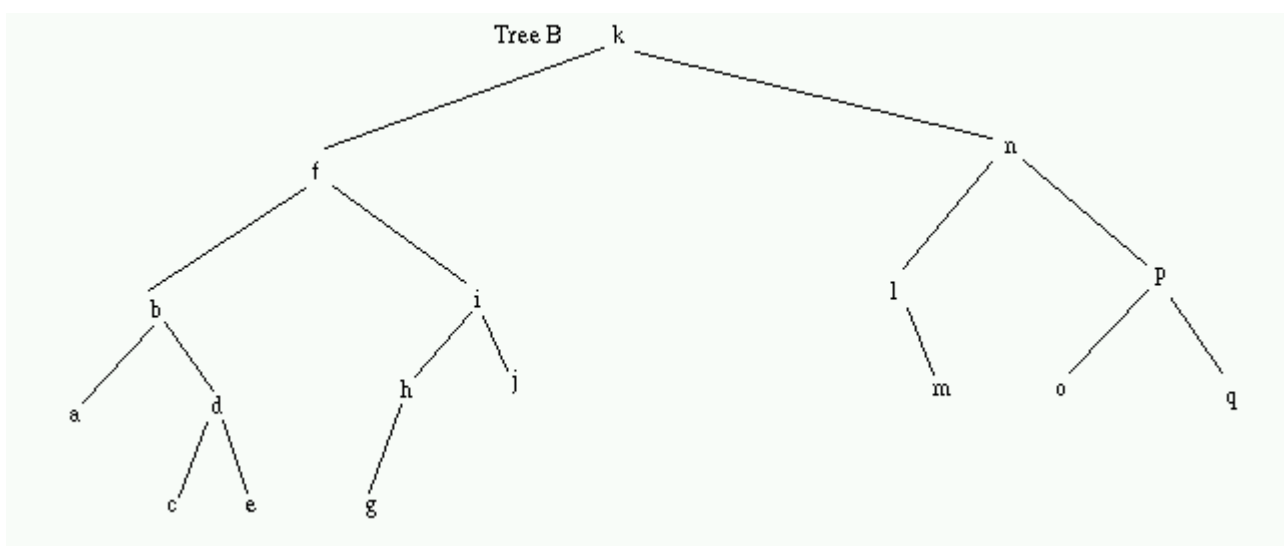
E S S A F O I M U I T O F A C I L

12. A ordem de inserção dos elementos da questão 4 produziu uma árvore balanceada? Caso contrário, qual seria uma ordem dos elementos que produziria uma árvore balanceada?

13. Suponha que uma certa Árvore Binária de Busca possua chaves com inteiros de 0 a 1000, e que executamos uma busca pelo valor 363. Qual das sequências apresentadas abaixo não seria possível de acontecer?

- 2, 252, 401, 398, 330, 363
- 2, 399, 387, 219, 266, 382, 381, 278, 363
- 923, 220, 911, 244, 898, 258, 362, 363
- 924, 278, 347, 621, 299, 392, 358, 363
- 925, 202, 910, 245, 363

14. Dada a seguinte Árvore Binária de Busca, responda



14.1 Qual é a altura da árvore?

14.2 Quais são os níveis correspondentes dos nós *a*, *f*, *g*, *k*, *l* e *p*?

14.3 A árvore está balanceada? Por que?

14.4 Como ficaria a árvore se um dos nós folha fossem removidos (*e.g. a, c, e, g, j, m, o, q*)? escreva uma função em C que executa tal tarefa.

14.5 Como ficaria a árvore ao remover o nó *l*? Escreva uma função em C que executa tal tarefa.

14.6 Como ficaria a árvore ao remover o nó *h*? Escreva uma função em C que executa tal tarefa.

14.7 Como ficaria a árvore ao remover qualquer um dos nós *b, d, f, i, k, n, p*? Escreva uma função que executa tal tarefa.

15. Escreva uma função que transforme um vetor crescente em uma árvore de busca balanceada. Escreva uma função que transforme uma árvore de busca em um vetor crescente. Comece por alocar dinamicamente o vetor.

16. Determinado sistema armazena registros por chaves numéricas em uma árvore AVL. Nessa árvore são inseridos os seguintes valores:

a) 50, 30, 20, 70, 40, 35, 37, 38, 10, 32, 45, 42, 25, 47, 36.

b) 100, 80, 60, 40, 20, 70, 30, 50, 35, 45, 55, 75, 65, 73, 77

nessa ordem. Apresente passo a passo (tanto para sequência a), quanto para sequência b)) como a árvore vai sendo construída. Realize as rotações necessárias e indique qual rotação foi realizada em cada caso.