

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
INF01124 – Classificação e Pesquisa de Dados - Semestre 2002/2

Prof. Manuel Menezes de Oliveira Neto

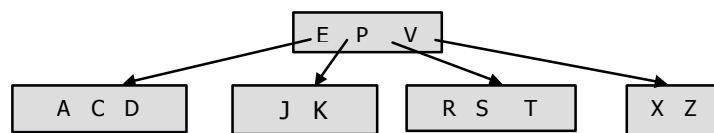
Lista de Exercícios Número 3

Data de Divulgação: 18/02/2003

Data de Entrega: 25/02/2003

- 1) Seja h uma função de hashing ideal, cujos resultados são distribuídos uniformemente ao longo de seu contra-domínio. Considere uma tabela T contendo m entradas, e seja $n > m$ o número de chaves (registros) a serem incluídas na tabela T , a qual se encontrava originalmente vazia. Qual o número de colisões esperado neste caso?
- 2) Insira os seguintes valores de chave em uma tabela contendo 9 entradas: 32, 49, 13, 27, 4, 96, 31, 25, 48. Mostre o resultado da tabela final.
 - a) Utilizando a função $h(c) = (c \bmod 9) + 1$ e resolvendo colisões utilizando *endereçamento aberto com realeatorização*.
 - b) Utilizando a função $h(c) = (c \bmod 8) + 2$ e resolvendo colisões utilizando *endereçamento aberto com busca linear*.
 - c) Utilizando a função $h(c) = (c \bmod 7) + 3$, utilizando *encadeamento com tabela de descritores*.
 - d) Compare o desempenho das três funções acima em termos do número de sinônimos que elas geram para os dados fornecidos. Explique este comportamento.
 - e) Considerando que o conjunto de chaves é fixo, crie uma função de hashing que minimize a ocorrência de colisões. (Nota: para o conjunto de chaves informado, é possível evitar colisões completamente).
- 3) Suponha a situação em que a resolução de colisões é feita por meio do uso de encadeamento e que as listas encadeadas sejam mantidas ordenadas pelo valor de chave. Discuta como esta organização afeta o tempo de execução dos algoritmos nos seguintes casos:
 - a) Busca por uma chave presente na tabela;
 - b) Busca por uma chave não-presente na tabela;
 - c) Inserção;
 - d) Exclusão
- 4) Utilize o método de *compressão de chaves alfanuméricas* para comprimir os seguintes valores de chave para 16 bits: ABACAXI, ABACATE, ABAJUR, ABANADOR. Os valores obtidos devem ser representados na base 10.

- 5) Repita o exercício da Questão 4, utilizando o *método da conversão de base* e interpretando o valor produzido pela operação XOR como sendo na base 11, e convertendo-o para a base 10.
- 6) Qual a diferença entre a *propriedade de uma árvore binária de pesquisa* e a *propriedade de um heap*?
- 7) Dado o conjunto de valores $C = \{1, 4, 7, 9, 16, 23, 25\}$ Construa árvores binárias de pesquisa com alturas (número máximo de arestas entre a raiz e um nó folha) iguais a 2, 3, 4, 5 e 6.
- 8) Considere o seguinte algoritmo de ordenação: dado um conjunto de valores a serem ordenados, construa uma árvore binária de pesquisa e imprima os valores dos nodos durante um percorrimento da árvore em pré-ordem. Discuta o custo deste algoritmo para o pior e melhor caso? Como ele se compara com os custos do Quicksort e do Merge Sort?
- 9) Mostre, passo a passo, a construção de uma árvore AVL a partir dos seguintes valores de chave, na ordem especificada, realizando as operações de rotação apropriadas quando necessário. Seqüência: 14, 6, 4, 2, 16, 20, 12, 13.
- 10) Seja A uma árvore B com *grau mínimo* $t = 3$ e altura (número de arestas entre o nodo raiz e as folhas) igual a 2.
- Qual o menor e qual o maior número de chaves que A pode conter em qualquer instante?
 - Expresse estes limites para uma árvore B genérica com grau mínimo $t = g$ e altura h .
- 11) Considere a seguinte árvore B com grau mínimo $t = 3$. Proceda as seguintes operações na ordem especificada e mostre a árvore resultante após a realização de cada operação:



- Remova C
- Remova F
- Insira B e G
- Remova X
- Insira Q e U