8ª Lista de Exercícios

Estrutura de Dados

Prof. Hamilton José Brumatto

Aplicações em Árvores Binárias

1. Considere a seguinte expressão: $34 \times (12 - 4)/2 * (4 \wedge (5/2)) - (4 + 3)$. Simule uma árvore de expressão para esta expressão, gere a expressão pós-fixa e calcule seu valor a partir daí. Compare o resultado calculando o valor direto da expressão.
2. Implemente uma estrutura de árvore de expressão, e duas rotinas, uma para inserir elementos usando expressão pos-fixa, e outro usando expressão infixa. A sua árvore deve calcular o resultado da expressão. (sugestão para facilitar a leitura, ao invés de escrever a expressão toda em uma linha, escreva na forma de coluna (várias linhas), assim o valor 34 não se confunde com 3 e 4 separados.)
3. Apresente um algoritmo que consiga inserir elementos em uma árvore de expressão usando uma expressão pré-fixa.
4. Considere o seguinte texto:
   "nas mensagens muito extensas contendo simbolos que aparecem raramente, a economia eh substancial; em geral, os codigos nao sao formados pela frequencia de caracteres dentro de uma unica mensagem isolada, mas por sua frequencia dentro de um conjunto inteiro de mensagens; o mesmo conjunto de codigos eh, entao, usado para cada mensagem."
   Simule a aplicação de um algoritmo de huffmann no texto e gere a tabela de códigos para os caracteres usados no texto. Qual seria o tamanho, em bits, do texto usando uma codificação formal de 7 bits (ASCII) e usando a compressão de Huffmann?
5. Construa uma árvore para realizar a compressão de Huffmann. Forneça o texto do item anterior e apresente uma tabela com os códigos binários de cada caracter. Se o texto for diferente a tabela será diferente? Por quê?
6. Já que a árvore binária de busca insere os elementos fazendo uma classificação, apresente uma proposta de algoritmo que utilize esta como fila de prioridade. Qual o custo (médio) para se inserir e remover elementos nesta fila?
7. Construa uma árvore binária de busca e apresente um algoritmo para ordenar uma sequência numérica usando a árvore binária de busca.
8. Construa uma árvore binária de busca que aceite as operações: I (inserir) e R (remover), onde cada operação recebe um valor a ser inserido ou removido. (Somente será removido se o valor existir na árvore, caso contrário será ignorado).