Curso de Tecnologia em Sistemas de Computação
Disciplina: Estrutura de Dados e Algoritmos
Gabarito da AP2 - Segundo Semestre de 2008

Nome -
Assinatura -

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
3. Você pode usar lápis para responder as questões.
4. Ao final da prova devolva as folhas de questões e as de respostas.
5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.

---

1. Valor (3.5)
Responder às perguntas abaixo, explicando o que você entende a respeito dos seguintes conceitos:

(a) Percurso em ordem simétrica de árvore binária.

Resposta: É uma visita sistemática a cada um dos nós da árvore binária, seguindo recursivamente os seguintes passos, para cada subárvore:
- percorrer sua subárvore esquerda, em ordem simétrica;
- visitar a raiz;
- percorrer sua subárvore direita, em ordem simétrica.

(b) Custo de árvore binária de busca.

Resposta: É o número total de comparações efetuadas, considerando-se buscas com e sem sucesso.

(c) Lista de prioridade.

Resposta: É uma tabela na qual a cada um de seus dados está associada uma prioridade.

(d) Árvore B.

Resposta: Seja d um número natural. Uma árvore B de ordem d é uma árvore ordenada que é vazia, ou que satisfaz as seguintes condições:
(i) a raiz é uma folha ou tem no mínimo dois filhos;
(ii) cada nó diferente da raiz e das folhas possui no mínimo $d+1$ filhos;
(iii) cada nó tem no mínimo $2d+1$ filhos;
(iv) todas as folhas estão no mesmo nível.

(e) Método de frequência de caracteres para compactação de dados.

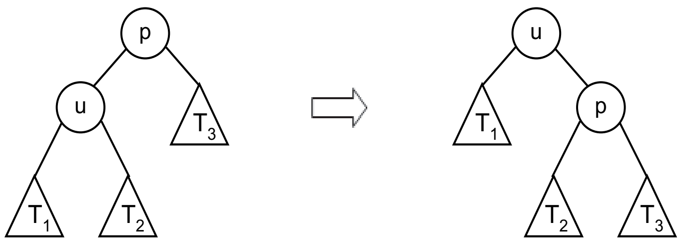
Resposta: Seja um arquivo composto de símbolos não numéricos. Neste método, determina-se a quantidade de símbolos idênticos consecutivos, existentes no texto. Então, substitui-se cada subcadeia de símbolos idênticos por um único símbolo, precedido da quantidade acima determinada. Se a quantidade de símbolos for igual a 1, por convenção, o valor numérico é omitido.

2. Valor (2.0)
   Descrever, inclusive através de desenhos, as operações de rotação esquerda e direita, e as operações de rotação dupla esquerda e direita. Explicar, com detalhe, a utilização das rotações nas operações de inclusão em árvores AVL.

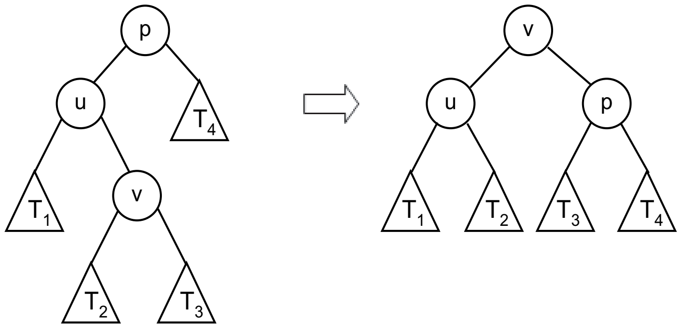
   Resposta: Após efetuar a inclusão de um nó q em uma árvore AVL, percorre-se o caminho ascendente que vai de q até a raiz, e verifica-se se existe algum nó p que se tornou desregulado (isto é, tal que a diferença de altura entre as duas subárvores de p tornou-se maior que um). Em caso afirmativo, podemos aplicar uma transformação apropriada para regulá-lo. Temos quatro casos, descritos a seguir. A notação para a compreensão dos casos é a seguinte: o nó u é o filho de p no caminho até q; $h_E(x)$ e $h_D(x)$ denotam as alturas das subárvores esquerda e direita do nó x, respectivamente.

   Caso 1: $h_E(p) > h_D(p)$.
   Então, q pertence à subárvore esquerda de p. Além disso, p possui o filho esquerdo $u \neq q$. Por isso, sabe-se que $h_E(u) \neq h_D(u)$. Temos duas possibilidades:

   Caso 1.1: $h_E(u) > h_D(u)$.
   Neste caso, q pertence a T_1. Observe que $h(T_1) - h(T_2) = 1$ e $h(T_2) = h(T_3)$. Logo, a aplicação da *rotação direita* da raiz p transforma a subárvore considerada da seguinte forma, restabelecendo a regulagem de p:

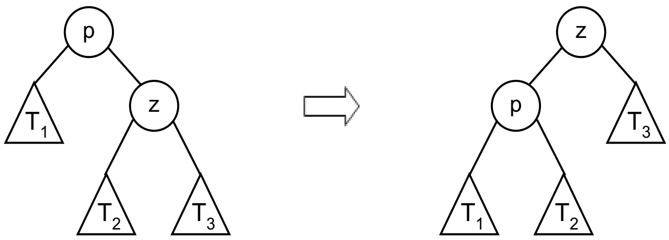


   Caso 1.2: $h_D(u) > h_E(u)$.
   Então u possui o filho direito v. Nesse caso, vale $|h(T_2) - h(T_3)| \leq 1$ e $\max\{h(T_2), h(T_3)\} = h(T_1) = h(T_4)$. Aplica-se então a *rotação dupla direita* da raiz p:

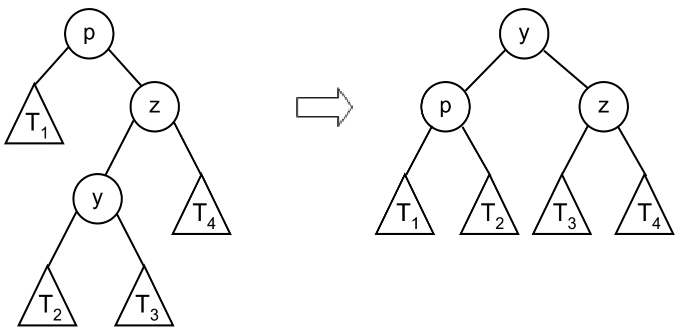


Caso 2: $h_D(p) > h_E(p)$.
Então, p possui o filho direito $z \neq q$. Segue-se que $h_E(z) \neq h_D(z)$. Temos duas possibilidades:

Caso 2.1: $h_D(z) > h_E(z)$.
Neste caso, q pertence a T_3. Temos $h(T_3) - h(T_2) = 1$ e $h(T_2) = h(T_1)$. Aplica-se então a *rotação esquerda*:



Caso 2.2: $h_E(z) > h_D(z)$.
Então z possui filho esquerdo y. As alturas de T_1, T_2, T_3 e T_4 satisfazem as mesmas relações do caso 1.2. Aplica-se então a *rotação dupla esquerda*:



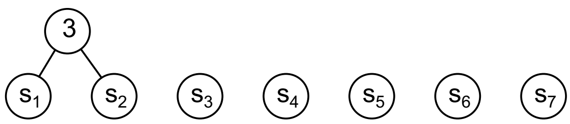
3. Valor (2.0)
   Aplicar o algoritmo de Huffman para os símbolos $\{s_1, \ldots, s_7\}$, onde o símbolo s_i possui freqüência i. Explicar cada um dos passos do algoritmo, separadamente, desenhando a floresta obtida, após cada passo. Quantos passos são realizados, no total ? No caso de aplicação de uma entrada de tamanho n, qual a complexidade total do algoritmo ?

   Resposta:

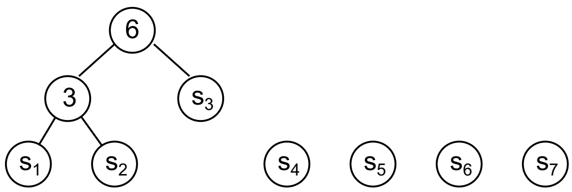
   Inicialmente, definimos 7 subárvores, cada qual consistindo de um único nó contendo o símbolo s_i, $1 \leq i \leq 7$:

   Em seguida, repetimos 6 vezes o seguinte passo:
   Escolher as duas subárvores T' e T'' de menor freqüência e substituí-las por $T' \oplus T''$. Os 6 passos são ilustrados a seguir:

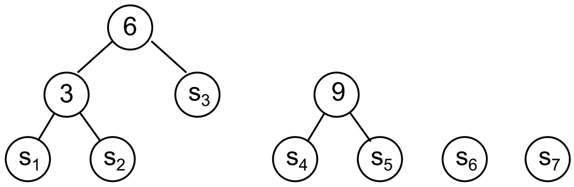
   Passo 1:



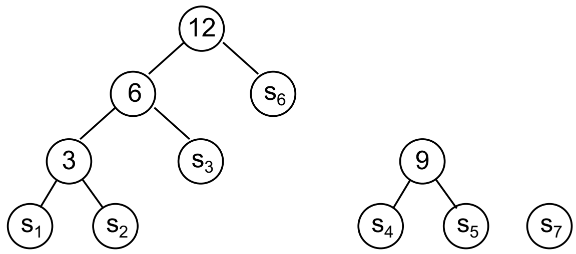
   Passo 2:



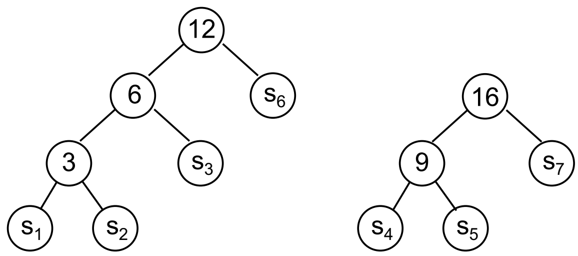
   Passo 3:



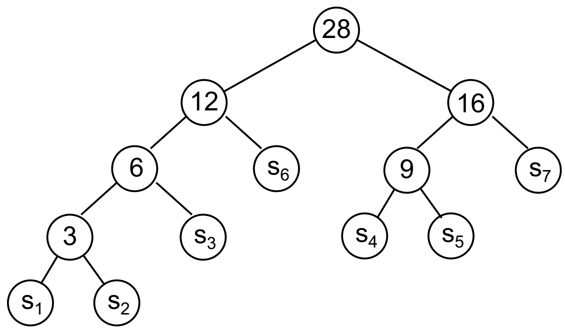
Passo 4:



Passo 5:



Passo 6:



Para este conjunto de símbolos, temos 6 passos realizados. Para uma entrada de tamanho n, seriam realizados $n - 1$ passos. Como cada passo possui complexidade $O(\log n)$, a complexidade total do algoritmo é $O(n \log n)$.

4. Valor (2.5)
   Deseja-se efetuar buscas em um arquivo contendo n registros, cada qual identificado por uma chave. Para tal, será utilizada uma tabela de dispersão, onde as chaves serão armazenadas. Qual deve ser a função de dispersão mais adequada, para atender às seguintes condições.

(a) Suponha que a chave de ordem i é igual a $3i$. Isto é, o conjunto das chaves é $\{3, 6, \ldots, 3n\}$. Nesse caso, escolher a função de dispersão de tal modo que (i) nunca ocorra colisão, e (ii) o tamanho da tabela de dispersão seja igual a n.

Resposta: $h(x) = x/3$.

(b)Suponha que a menor chave seja igual a 1, a maior seja igual a $3n$ e que as demais possam assumir quaisquer valores entre 1 e $3n$. Nesse caso, escolher a função de dispersão de tal modo que (i) nunca ocorra colisão, e (ii) o tamanho da tabela não seja maior do que $3n$.

Resposta: $h(x) = x$.