

Esta lista pode ser desenvolvida em grupos de até dois alunos, seguindo as especificações contidas no arquivo 00_ProcListas.pdf disponível na área pública da disciplina na rede. A entrega ao professor deverá ocorrer até o dia **12 de dezembro de 2018**.

- 1) Adapte o programa que implementa a árvore binária ordenada de inteiros apresentada na apostila, para que ele imprima na tela o conteúdo de cada nó e, após isso, também exiba:
 - Maior valor contido na árvore.
 - Menor valor contido na árvore.
 - Quantidade de folhas da árvore.
 - A altura da árvore.
 - Pesquisa na árvore de um valor informado pelo usuário.

Obs: Criar uma função para realizar cada um dos processamentos solicitados acima.

- 2) Dadas duas árvores binárias A e B, diz-se que $A \text{ eq } B$ (lê-se A é equivalente a B) se:
 - a. ambas são vazias, ou
 - b. $\text{info}(\text{raiz}(A)) = \text{info}(\text{raiz}(B))$ e $\text{esq}(A) \text{ eq } \text{esq}(B)$ e $\text{dir}(A) \text{ eq } \text{dir}(B)$.
 Faça um programa que permite montar duas árvores binárias ordenadas de valores inteiros e determinar se elas são equivalentes.
- 3) Segundo as páginas 187 e 188 da Introdução a Estruturas de Dados, de Waldemar Celes e outros, “Para descrever árvores binárias, podemos usar a seguinte notação textual: a árvore vazia é representada por $\langle \rangle$, e árvores não-vazias, por $\langle \text{raiz } \text{sae } \text{sad} \rangle$. Com essa notação, a árvore da Figura 13.4 é representada por:

$\langle a \langle b \langle \rangle \rangle \langle c \langle e \langle \rangle \rangle \langle f \langle \rangle \rangle \rangle \rangle$

Pela definição, uma subárvore de uma árvore binária é sempre especificada como sendo a *sae* ou a *sad* de uma árvore maior, e qualquer das duas subárvores pode ser vazia. Assim, as duas subárvores da Figura 13.5 são distintas.

Isso também pode ser visto pelas representações textuais das duas árvores que, em **pre-order**, são, respectivamente: $\langle a \langle b \langle \rangle \rangle \langle c \rangle \rangle$ e $\langle a \langle \rangle \langle b \langle \rangle \rangle \langle c \rangle \rangle$.”

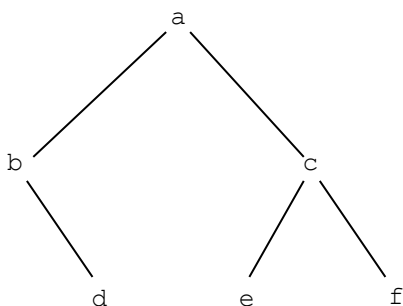


Figura 13.4 Exemplo de árvore binária.

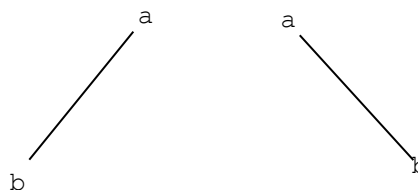


Figura 13.5 Duas árvores binárias distintas

Faça um programa que recebe uma sequência de inteiros e monta a árvore binária ordenada correspondente, encerrando a entrada de dados quando o valor -999 for informado. Em seguida exibir a descrição da árvore na notação textual indicada anteriormente, porém em sentido **infixo**.

- 4) Faça uma versão do programa da lista de cidades criado no exercício 3 da lista 1 com a implementação sendo por contiguidade. Cada cidade terá como informações úteis o seu nome, população, área territorial, PIB e IDH, sendo que estes 3 últimos dados são números reais de precisão simples. Considere que poderemos ter até 5000 cidades diferentes e, após montada a lista, o programa deverá receber um inteiro N (entre 1 e 50) correspondente à quantidade de cidades que serão beneficiadas por uma iniciativa do governo direcionada para as localidades de menor IDH. Em seguida, imprimir na tela todos os dados dos N municípios selecionados. Utilize

Lista de Exercícios N° 2

uma fila de prioridades montada por meio de um MIN-HEAP, com base nos programas de exemplo disponibilizados pelo professor na rede.