DCC–IME–USP

Routo Terada
Sala 204C Tel.: 3091 5918
e-mail rt@ime.usp.br

MAC5710 **Estrutura de Dados**

Primeiro Semestre de 2004

Exercício-Programa 2 Data de entrega: até **16 de junho de 2004.**

Árvores 2-3

Este exercício é para ser elaborado em linguagem C/ANSI. O objetivo é analisar estatisticamente as Árvores 2-3. A definição de tais árvores foi dada em aula, e não deve ser adotada qualquer outra definição.

As funções solicitadas abaixo podem possuir outros parâmetros de entrada ou de saída.

1. Escrever uma função – Buscar(A, x) – que procura a chave inteira x na árvore 2-3 A.
2. Escrever uma função – Incluir(A, x) – que inclue a chave inteira x na árvore 2-3 A. Os discriminadores $p.q$ em cada nó interno devem ser atualizados! Cada folha deve conter apenas **uma** chave inteira. Pode ocorrer repetição de chaves. Uma sugestão para esta função (que pode ser aceita ou não): chamar a função Buscar(A, x)
3. Escrever uma função – Altura(A) – que calcula a altura da árvore 2-3 A.
4. Escrever uma função - GeraInt(a, b) – que gera uma chave inteira distribuída uniformemente no intervalo $[a, b]$, utilizando a função $rand()$ e $srand()$ da biblioteca `stdlib.h`

Escrever um programa que utilize necessariamente as funções solicitadas.

O intervalo $[a, b]$ deve ser $[0, 100]$. Existem dois parâmetros globais: $Narv$ e $Nchave$. Você deve gerar $Narv$ árvores 2-3 A, uma de cada vez. No início, cada árvore deve ser vazia; cada uma das $Nchave$ chaves é inserida, uma a uma, consecutivamente.

Os seus testes deverão ser para os valores $Narv = 50$ e $Nchave = 200$.

Apenas para a **primeira árvore**, e apenas para as primeiras 50 chaves incluídas, o programa deve, após cada **dez** inclusões de chave:

1. mostrar na tela os discriminadores p, q e o conteúdo de todas as folhas (com pausa apropriada após um número adequado de linhas)
2. calcular e mostrar a altura da árvore parcial.

Após construir cada árvore, inclusive a primeira, o programa deve:

1. mostrar na tela as primeiras 50 chaves nas folhas, em ordem crescente (com pausa apropriada após um número adequado de linhas)

2. calcular e mostrar a altura da árvore construída.

A memória utilizada para a árvore deve ser reutilizada para a construção da próxima árvore.

O programa não possuirá qualquer dado de entrada ou qualquer arquivo de saída em disco. Após uma única execução do programa, a saída final na tela deverá conter pelo menos:

1. os valores $Narv, Nchave$ utilizadas

2. a **média** e o **desvio padrão** das alturas das árvores. Observe que, para calcular a média e o desvio padrão, não há necessidade de armazenar todos os valores da altura.

Observações

1. A nota atribuída ao exercício-programa também vai depender dos seguintes fatores: qualidade da decomposição do programa em funções e procedimentos; qualidade e clareza dos algoritmos; qualidade das estruturas de dados; qualidade da listagem; qualidade do layout, da ortografia e da pontuação; qualidade dos nomes das variáveis e tipos. Exercícios-programas com erro de sintaxe receberão nota zero.

2. Este exercício é para ser feito *individualmente*.

3. Coloque comentários em seu programa explicando o que cada etapa do programa significa! Isso será levado em conta na sua nota.

4. Coloque como comentário o seu nome, número USP, qual o compilador, qual o sistema operacional e qual o modelo de computador que V usou.

5. Faça uma saída clara! Isso será levado em conta na sua nota.

6. Não deixe para a última hora. Planeje investir 70 porcento do tempo total de dedicação em escrever o seu programa todo e simular o programa SEM computador (eliminando erros de lógica) ANTES de digitar e compilar no computador. Isso economiza muito tempo e energia.