

Universidade Federal da Bahia
Graduação em Ciência da Computação
MATA52 - Análise e Projeto de Algoritmos
Primeiro Trabalho Prático
Prof. Flávio Assis
Semestre 2015.2 - 28 de março de 2016

Programação Dinâmica

1 Descrição Geral do Trabalho

Neste trabalho o aluno fará uma implementação do algoritmo para encontrar uma árvore de busca binária ótima, descrito no livro de Cormen et al. *Introduction to Algorithm*. O programa receberá como entrada sequências de chaves e de probabilidades de acesso e retornará como saída uma árvore de busca binária ótima.

2 Formato de Entrada e Saída

A entrada para o programa terá quatro linhas. A primeira linha conterá um número n . A segunda linha conterá uma sequência de n chaves inteiras distintas, k_1, k_2, \dots, k_n , em ordem crescente ($k_1 < k_2 < \dots < k_n$). A terceira linha conterá n valores de probabilidade, p_1, p_2, \dots, p_n . O valor p_i corresponde à probabilidade de acesso à chave k_i , $1 \leq i \leq n$. A quarta linha conterá $n + 1$ valores de probabilidade q_0, q_1, \dots, q_n . O valor q_0 representa a probabilidade de se consultar uma chave menor que k_1 . O valor q_n representa a probabilidade de se consultar uma chave maior que k_n . Os demais valores q_i , $1 \leq i < n$, representam a probabilidade de se consultar uma chave entre k_i e k_{i+1} .

A saída consistirá de uma representação da árvore binária, como descrita a seguir. Como descrito no livro, os valores entre chaves serão representados por *chaves auxiliares*, representadas pela sequência d_0, d_1, \dots, d_n . A representação da árvore deve ser feita listando os valores referentes a cada nó, um por linha, seguindo a ordem de visita de um percurso *em ordem* da árvore. Para cada nó, os dados devem ser apresentados no seguinte formato: a sequência de caracteres '*chave:*', seguida de um espaço, seguido do valor da chave armazenada no nó, seguido de um espaço, seguido pela sequência de caracteres '*fesq:*', seguida por um espaço, seguido pelo valor da chave armazenada no filho à esquerda, seguido por um espaço, seguido pela sequência de caracteres '*fdir:*', seguida por um espaço, seguido pelo valor da chave armazenada no filho à direita. Caso não haja filho à esquerda ou filho à direita, a sequência de caracteres *nil* deve ser gerada, no local correspondente à chave do filho. As chaves auxiliares d_i devem

ser representadas pela letra d seguida imediatamente pelo índice. Por exemplo, d_4 deve ser representada por d4, d_{10} por d10, etc.

Importante: o programa não deve gerar nenhum caractere a mais na saída, além dos indicados acima. Em particular, o programa não deve conter menus.

Não deve haver espaço entre linhas na saída. A saída deve apresentar os caracteres em letras minúsculas.

3 Observações

Trabalho em equipes de até três alunos. A nota de cada aluno, no entanto, dependerá da nota obtida em uma avaliação escrita, em sala, sobre o trabalho. A avaliação será individual.

Data de entrega: 19/04/2016

Linguagens de programação permitidas: C, C++, Java ou Python.

Observação Importante: Para as linguagens C, C++ e Java, somente trabalhos feitos utilizando-se os seguintes compiladores serão aceitos:

- C: gcc ou djgpp
- C++: g++ ou djgpp
- Java: compilador java recente, disponibilizado pela Oracle.

Não serão compilados trabalhos em outros compiladores! Erros ocasionados por uso de diferentes compiladores serão considerados erros do trabalho!

A entrega do trabalho deverá ser feita através do moodle, por um dos alunos da equipe. O trabalho deve conter claramente os nomes dos alunos autores do trabalho. O arquivo submetido deve ser protegido por senha. A senha e os nomes dos participantes da equipe devem ser enviados por email ao professor (o aluno deve utilizar *zip*).