



Curso: Engenharia de Computação
Disciplina: Algoritmos e Estruturas de Dados I
Professor: Alexandre Magno de Sousa

1º semestre de 2023

Data: 25/07/23

Data de entrega: 11/08/23

TAD ÁRVORES BINÁRIAS DE BUSCA

1. Desenhe árvores de busca binária de alturas 2, 3, 4, 5 e 6 para o conjunto de chaves $\{1, 4, 5, 10, 16, 17, 21\}$.
2. Apresente a impressão dos elementos da árvore binária da questão anterior com altura 2 para: (a) caminhamento central; (b) caminhamento em pré-ordem; e, por fim, (c) caminhamento em pós-ordem.
3. Suponha que temos números entre 1 e 1000 em uma árvore de busca binária e que queremos procurar o número 363. Qual das seguintes sequências **não** poderia ser a sequência de nós examinados?
 - (a) 2, 252, 401, 398, 330, 344, 397, 363.
 - (b) 924, 220, 911, 244, 898, 258, 362, 363.
 - (c) 925, 202, 911, 240, 912, 245, 363.
 - (d) 2, 399, 387, 219, 266, 382, 381, 278, 363.
 - (e) 935, 278, 347, 621, 299, 392, 358, 363.
4. Sejam p_1 e p_2 duas permutações de um conjunto de n chaves $S = \{s_1, s_2, \dots, s_n\}$ e T_1 e T_2 as duas árvores binárias de pesquisa geradas a partir de p_1 e p_2 , respectivamente. Cada árvore é gerada a partir da inserção sequencial das chaves de sua permutação iniciando pela chave mais à esquerda. Avalie as seguintes afirmações:
 - I. $T_1 \neq T_2$ se e somente se $p_1 \neq p_2$.
 - II. As ordens de impressão das chaves de T_1 e T_2 obtidas por um caminhamento central (também chamado de caminhamento in-ordem) podem ser distintas.
 - III. É sempre possível saber em tempo logarítmico se um elemento $s_i \in S$ está na árvore T_1 .
 - IV. Seja h_1 a altura da árvore T_1 . A complexidade de tempo para se encontrar a chave sucessora de uma chave s_i qualquer em T_1 é $O(h_1)$ (no caso, chave sucessora é a chave seguinte à s_i quando elas estão ordenadas crescentemente).

Estão corretas apenas o que se afirma em

- (a) I, II e III.
 - (b) I, III e IV.
 - (c) I e III.
 - (d) II e IV.
 - (e) II, III e IV.
5. A operação de eliminação é “*comutativa*” no sentido de que eliminar x e depois y de uma árvore de busca binária resulta na mesma árvore que eliminar y e depois x ? Mostre por que ou dê um contraexemplo.
 6. Apresente uma versão iterativa para as funções de percurso in-ordem, percurso em pré-ordem, percurso em pós-ordem.
 7. Apresente a versão iterativa para a operação INSERE.
 8. Dê um algoritmo iterativa que execute um percurso de árvore in-ordem de maneira que a função deve utilizar o TAD Pilha como estrutura de dados auxiliar.
 9. Um método alternativo de se executar um percurso de árvore em ordem em uma árvore de busca binária de n nós encontra o elemento mínimo na árvore chamando a função MINIMO e depois fazendo $n - 1$ chamadas a função SUCESSOR. Mostre que o algoritmo é executado em tempo $\Theta(n)$.
 10. Podemos ordenar um dado conjunto de n números construindo primeiro uma árvore de busca binária contendo esses números (usando INSERE repetidamente para inserir os números um a um) e então imprimindo esses números por um percurso de árvore em ordem. Quais são os tempos de execução do pior caso e do melhor caso para esse algoritmo de ordenação?