

Drzewa poszukiwań binarnych (BST)

Zad. 1. Naucz się algorytmu usuwania wierzchołków z drzewa poszukiwań binarnych opisanego w książce „Wprowadzenie do algorytmów” Cormena, Leisersona i Rivesta.

Zad. 2. Napisz program rekonstruujący dane drzewo BST do postaci pełnego drzewa BST, czyli takiego w którym liście znajdują się na dwóch sąsiednich poziomach w drzewie.

Zad. 3. Napisz funkcję SZUKAJ sprawdzającą, czy podany klucz znajduje się w drzewie BST.

Zad. 4. Przyjmijmy, że w drzewie BST znajdują się liczby od 1 do 1000 i chcemy wyszukać liczbę 363. Które z poniższych ciągów wierzchołków nie mogą zostać sprawdzone w procedurze SZUKAJ?

- 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.

Zad. 5. Dla każdego z następujących ciągów kluczy narysuj drzewo BST uzyskane przez wstawienie kluczy, jeden po drugim, w danym porządku, do początkowo pustego drzewa.

- a) 1, 2, 3, 4, 5, 6, 7.
- b) 4, 2, 1, 3, 6, 5, 7.
- c) 1, 6, 7, 2, 4, 3, 5.

Zad. 6. Dla każdego drzewa BST uzyskanego w zadaniu 5. narysuj drzewo po usunięciu korzenia.

Zad. 7. Opisz, jak mógłbyś obliczyć wysokość drzewa poszukiwań binarnych, zbudowanego z podanego zestawu kluczy, bez budowania drzewa BST.