

Struktury danych i złożoność obliczeniowa

Laboratorium 3 – Drzewo poszukiwań binarnych

Celem zadania jest implementacja słownika w postaci drzewa poszukiwań binarnych (**BST**), tzn. takiego, w którym ignoruje się problem wyważania drzewa.

Każdy z węzłów drzewa posiada następujące pola:

- klucz typu całkowitego (**int**),
- wskaźniki do lewego i prawego potomka (**wskaźnik do rodzica jest niedozwolony!**),
- 10-elementowa tablica znaków (**char**).

Należy stworzyć funkcje realizujące:

- inicjację pustego drzewa BST;
argumenty: opcjonalnie wskaźnik na korzeń drzewa;
- wstawienie do drzewa nowego elementu o zadany klucz; w tablicy znaków umieszczamy tekstową reprezentację wartości klucza, np. dla klucza o wartości **1123** w tablicy powinien pojawić się łańcuch „1123”; nie dopuszczamy do wielokrotnego wstawiania elementów o tym samym kluczu;
argumenty: opcjonalnie wskaźnik na korzeń drzewa, obowiązkowo klucz;
- wstawienie do drzewa **X** nowych węzłów o wygenerowanych losowo kluczach z zakresu -10000 do 10000;
argumenty: opcjonalnie wskaźnik na korzeń drzewa, obowiązkowo wartość X;
- wyszukanie w drzewie elementu o podanym jako argument kluczu (wraz z obsługą przypadku, w którym element nie istnieje);
argumenty: opcjonalnie wskaźnik na korzeń drzewa, obowiązkowo klucz;
- usunięcie z drzewa elementu o podanym jako argument kluczu (wraz z obsługą przypadku, w którym element nie istnieje); w przypadku usuwania węzła stopnia 2-go należy zaimplementować **wybraną przez siebie** wersję postępowania, odwołującą się do poprzednika albo następnika usuwanego węzła;
argumenty: opcjonalnie wskaźnik na korzeń drzewa, obowiązkowo klucz;
- przejście drzewa i wyświetlanie jego elementów w trybach **preorder**, **inorder** oraz **postorder** (trzy osobne funkcje); funkcje mogą być zaimplementowane z wykorzystaniem rekurencji; każda z funkcji wyświetla dodatkowo na końcu informację o liczbie odwiedzonych węzłów;
argumenty: opcjonalnie wskaźnik na korzeń drzewa.

Drzewo można zaimplementować strukturalnie (C) lub obiektowo (C++). W przypadku implementacji strukturalnej wskaźnik na korzeń drzewa powinna zostać przekazany do implementowanych funkcji jako argument (korzeń **nie może** być zmienną globalną).

Funkcje wstawiania, wyszukiwania i usuwania nie mogą wykorzystywać rekurencji.

Wskazówka: funkcja wyszukiwania może zwracać stos ze wskaźnikami na węzły rodzicielskie.

Program po uruchomieniu wczytuje plik wejściowy `inlab03.txt`.

Plik `inlab03.txt` zawiera w pierwszej linii kolejno liczbę elementów do wylosowania **X**, a następnie wartości czterech kluczy **k1**, **k2**, **k3**, **k4**.

Następnie wywoływana jest sekwencja funkcji (dalej w funkcji `main()`):

- czas start;
- zainicjuj „puste” drzewo;
- usuń element o wartości klucza **k1**;
- wstaw element o wartości klucza **k1**;

- wstaw **X** elementów do drzewa;
- wyświetl wszystkie klucze w trybie **inorder**;
- wyświetl wszystkie klucze w trybie **preorder**;
- wstaw element o wartości klucza **k2**;
- wyświetl wszystkie klucze w trybie **inorder**;
- wstaw element o wartości klucza **k3**;
- wstaw element o wartości klucza **k4**;
- usuń element o wartości klucza **k1**;
- wyświetl wszystkie klucze w trybie **preorder**;
- wyszukaj element o wartości **k1**;
- usuń element o wartości klucza **k2**;
- wyświetl wszystkie klucze w trybie **inorder**;
- usuń element o wartości klucza **k3**;
- usuń element o wartości klucza **k4**;
- czas stop;
- wypisz czas wykonania.

Przygotowanie e-maila do wysłania:

Uwaga! Kod źródłowy programu (1 plik) po oddaniu prowadzącemu zajęcia laboratoryjne musi zostać przesłany na adres `sdizo@zut.edu.pl` :

- plik z kodem źródłowym musi mieć nazwę: `nr_albumu.sdizo.lab03.main.c` (np. `123456.sdizo.lab03.main.c`); jeśli kod źródłowy programu składa się z wielu plików, to należy stworzyć jeden plik, umieszczając w nim kody wszystkich plików składowych; (plik może mieć rozszerzenie `.c` lub `.cpp`)
- plik musi zostać wysłany z poczty ZUT (`zut.edu.pl`);
- nagłówek maila (temat) musi mieć postać: `SDIZO I1 XXXY LAB03`, gdzie `XXXY` to numer grupy (np. `SDIZO I1 210C LAB03`);
- w pierwszych trzech liniach pliku z kodem źródłowym w komentarzach muszą znaleźć się:
 - linia 1: informacja identyczna z zamieszczoną w nagłówku maila
 - linia 2: imię i nazwisko
 - linia 3: adres email
- email **nie** powinien zawierać żadnej treści (tylko załącznik).