

Struktury danych i złożoność obliczeniowa

Laboratorium 6 – Kopiec sprzężony

Zadanie

Celem zadania jest implementacja kopca sprzężonego.

Każdy z węzłów drzewa posiada:

- klucz typu integer;
- referencje do lewego i prawego potomka (wskaźnik do rodzica jest niedozwolony);
- tablicę znaków mieszczącą 16 elementów;

Należy stworzyć funkcje realizujące:

- inicjację pustego kopca sprzężonego (z korzeniem);
- wstawienie do kopca nowego elementu, funkcja tablicy znaków przypisuje losowe wartości (losowe znaki);
- wstawienie do drzewa X nowych węzłów o wygenerowanych losowo i różnych wartościach składowych kluczowych z zakresu 25 do 30000 (wartość X podana jako argument funkcji);
- usunięcie z kopca sprzężonego elementu o:
 - najmniejszej wartości składowej kluczowej;
 - największej wartości składowej kluczowej.

Kopiec można zaimplementować strukturalnie (C) lub obiektowo (C++);

UWAGA: Niedopuszczalna jest implementacja tablicowa. Dopuszczalna jest wymiana wartości składowych kluczowych między węzłami w procesie przywracania porządku kopcowego.

Na liście argumentów każdej z wymienionych funkcji mogą się pojawić wyłącznie:

- referencja (wskaźnik) do korzenia drzewa (tylko w przypadku implementacji strukturalnej);
- wartość składowej kluczowej – za wyjątkiem funkcji wstawiania X elementów, która zamiast wartości składowej kluczowej zawiera liczbę elementów do wylosowania (w przypadku implementacji strukturalnej i obiektowej).

Program po uruchomieniu wczytuje plik wejściowy `inlab06.txt`.

Plik `inlab06.txt` zawiera w pierwszej linii liczbę: **X1**.

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

- czas start;
- zainicjuj kopiec sprzężony;
- wstaw **X1** elementów do kopca;
- wyświetl klucze wszystkich węzłów drzewa w porządku „preorder”;
- usuń z kopca element o kluczu najmniejszym, jednocześnie wyświetlając wartość tego klucza na konsoli operatorskiej;
- usuń z kopca element o kluczu największym, jednocześnie wyświetlając wartość tego klucza na konsoli operatorskiej;
- usuń z kopca wszystkie węzły wywołując naprzemiennie operacje usuwania elementu o kluczu największym i najmniejszym i wyświetlając wartość klucza usuwanego węzła;
- 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.lab06.main.c` (np. `123456.sdizo.lab06.main.c`); jeśli kod źródłowy programu składa się z wielu plików

(a także jednego lub więcej, ale z rozszerzeniem .cpp), to należy stworzyć jeden plik, umieszczając w nim kody wszystkich plików składowych;

- plik musi zostać wysłany z poczty wydziałowej (wi.zut.edu.pl);
- nagłówek maila (temat) musi mieć postać: SDIZO IS1 XXXY LAB06, gdzie XXXY to numer grupy (np. SDIZO IS1 210C LAB06);
- 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).