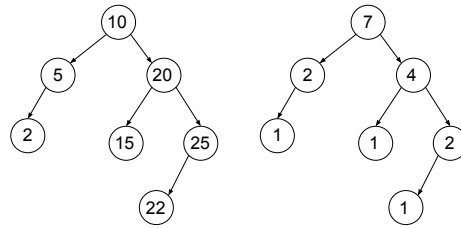


ЗАДАЧИ ЗА ЗАДЪЛЖИТЕЛНА  
САМОПОДГОТОВКА  
ПО  
Структури от данни и програмиране  
*Двоични дървета 3*

*email: kalin@fmi.uni-sofia.bg*

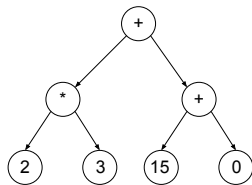
3 ноември 2016 г.

1. Дадено е двоично дърво. Да се напише функция, проверяваща дали има поне две различни нива от дървото, чиито множества от елементи съвпадат.
2. Да се дефинира метод на клас `BTree<T>`  
`BTree<T> deletedBOT (const T& x) const,`  
който построява копие на двоичното дърво с премахнат елемента със стойност `x`. Да се приеме, че дървото е наредено. Операцията да се извърши чрез единствен частен статичен метод на класа  
`private: static Node<T>* deleted (const T&),`  
който не използва други методи на класа, освен метода `BTree<T>::minimal`.  
Упътване: реализирайте `BTree<T>::deleted` по подобие на метода `BTree<T>::insertedBOT`.
3. Да се реализира метод `bool BTree<T>::isBOT()`, който проверява дали двоичното дърво е наредено.



Фигура 1. Примерно дърво и същото дърво, стойностите на чиито възли са заместени с размера на съответното им поддърво.

4. Стойността на всеки възел  $V$  в дадено двоично дърво от числа да се замени с броя на всички елементи на поддървото, на което  $V$  е корен. Вж. фигура 1. При операцията всеки от възлите да бъде посетен най-много веднъж.



Фигура 2. Примерно дърво с аритметичен израз.

5. Нека е дадено двоично дърво с елементи от тип `char`, за което е изпълнено, че:
- Дървото е непразно
  - Всеки от възлите му има точно 2 или 0 наследника
  - Елементите с 2 наследника съдържат един от символите `+`, `-`, `*` и `/`
  - Елементите с 0 наследника съдържат цифра

Да се дефинира функция, която връща стойността на аритметичния израз, съответен на дървото. Например, стойността на израза, съответен на дървото от фигура 2, е 21.

- б) Функцията да генерира `assertion failure`, ако дървото не отговаря на някое от описаните условия.