

[2pkt.] **Zadanie 2.**

Szablon rozwiązania: zad2.py

Dane jest drzewo binarne T , gdzie każda krawędź ma pewną wartość. Proszę zaimplementować funkcję:

```
def valuableTree(T, k):  
    ...
```

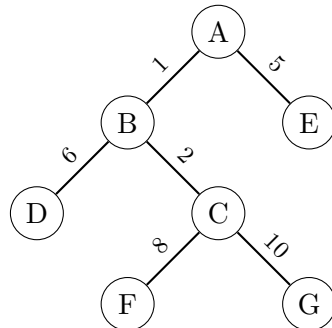
która zwraca maksymalną sumę wartości k krawędzi tworzących spójne poddrzewo drzewa T . Funkcja powinna być jak najszybsza. Proszę oszacować złożoność czasową oraz pamięciową zastosowanego algorytmu.

Drzewo T reprezentowane jest przez obiekty klasy `Node`:

```
class Node:  
    def __init__(self):  
        self.left      = None # lewe poddrzewo  
        self.leftval   = 0    # wartość krawędzi do lewego poddrzewa jeśli istnieje  
        self.right     = None # prawe poddrzewo  
        self.rightval  = 0    # wartość krawędzi do prawego poddrzewa jeśli istnieje  
        self.X         = None # miejsce na dodatkowe dane
```

Pole X można wykorzystać do przechowywania dodatkowych informacji w trakcie obliczeń.

Przykład. Rozważmy następujące drzewo:



Wywołanie `valuableTree(A, 3)` powinno zwrócić wartość 20, odpowiadającą krawędziom B-C, C-F i C-G.