15 DP on Trees

Problem 1: Maximum Sum Path

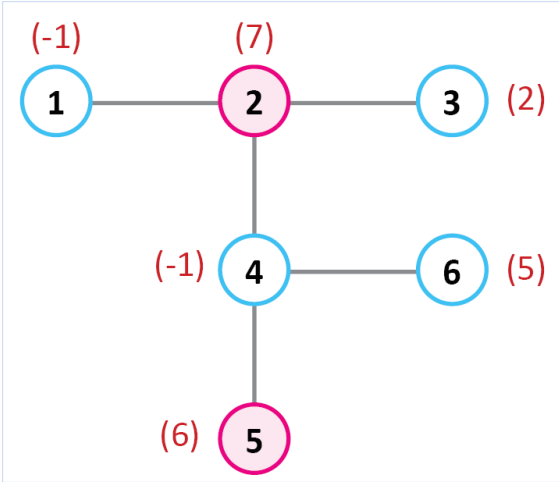
Given a tree T of N (1 <= N <= 100,000) nodes, where each node i has a value Vi (-10,000 <= Vi <= 10,000). You have to find a path between root and another node such that sum of values of the chosen nodes on the path is the largest when no two adjacent nodes (i.e. nodes connected directly by an edge) are chosen. The length of the path can be 0 and the node 1 is the root node.

Sample Input

```
6
-1 7 2 -1 6 5
1 2
3 2
4 6
5 4
2 4
```

Sample Output

```
13
```



Exercise 1: Maximum Sum Path

In the Maximum Sum Path problem we use the solution of each subtree only once to calculate its parent's solution. Therefore we do not have to save sub-solutions.

Re-implement the Maximum Sum Path program without using the memoization arrays SolIn[] and solOut[].

Your recursion function should return a pair of integers as the solution of the current subtree.

Problem 2: Maximum Sum Path 2

Given a tree T of N (1 <= N <= 100,000) nodes, where each node *i* has a value Vi (-10,000 <= Vi <= 10,000). You have to find a path between any two nodes such that sum of values of the chosen nodes on the path is the largest when no two adjacent nodes (i.e. nodes connected directly by an edge) are chosen. The length of the path can be 0.

Sample Input

9

6 -1 -2 9 3 -1 7 8 2

1 2

3 2

2 8

3 4

4 5

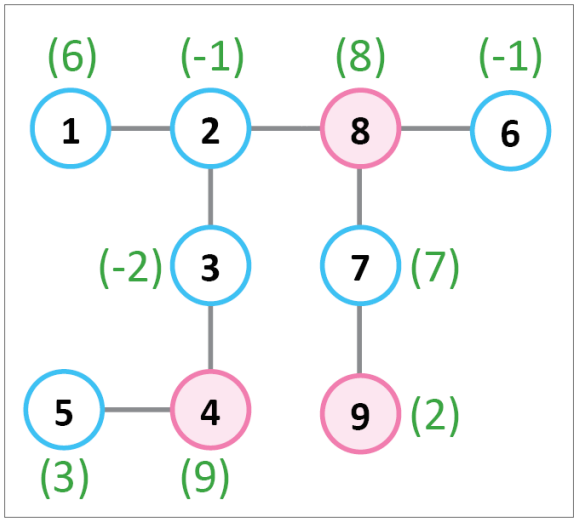
6 8

7 8

7 9

Sample Output

19



Problem 3: Maximum Sum Subset

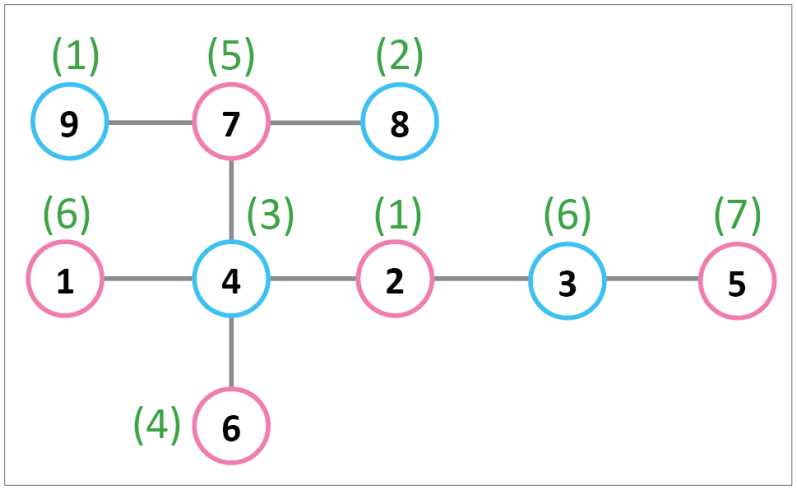
Given a tree T of N (1 <= N <= 100,000) nodes, where each node i has a value Vi (1 <= Vi <= 10,000). You have to choose a subset of nodes such that no two adjacent nodes (i.e. nodes connected directly by an edge) are chosen and sum of values of the nodes in chosen subset is maximum.

Sample Input

```
9
6 1 6 3 7 4 5 2 1
2 3
2 4
5 3
6 4
7 4
4 1
8 7
7 9
```

Sample Output

```
23
```



Problem 4: K-Leaf Tree

You are given an unweighted, undirected tree with N (1 <= N <= 100,000) nodes. There is a weight wi (1 <= wi <= 1000) assigned to each node. The aim is to delete enough nodes from the tree so that the tree is left with precisely K (0 <= K <= 1000) leaves.

The cost of such a deletion is the sum of the weights of the nodes deleted. After deleting some nodes, tree still should be connected.

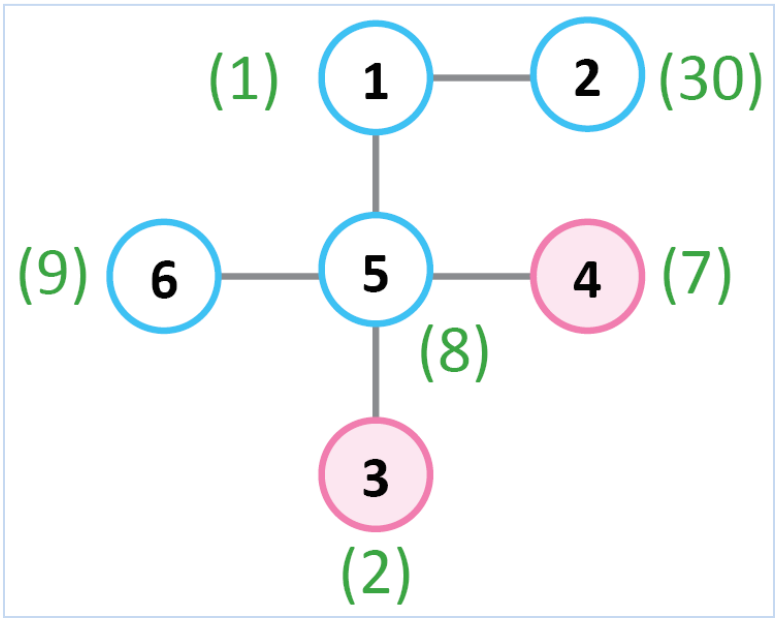
What is the minimum cost to reduce the given tree into a tree with K leaves?

PS: Root is not considered as a leaf node.

K-Leaf Tree (Continue)

Sample Input

```
6 2 1        //N, K, root
1 30 2 7 8 9
1 2
1 5
5 3
5 4
5 6
```



Sample Output

```
9
```