

Problem 34

В задаче на каждую вершину накладывается два условия:

- 1) Расстояние до любой другой вершины меньше k .
- 2) Нет полупути, который имеет длину k и проходит через эту вершину.

Второе условие покрывает первое, т.е. если для вершины a есть вершина b такая, что расстояние от a до b равно k , то существует полупуть длины хотя бы k (который проходит через a, b).

Таким образом, для каждой вершины надо проверить есть ли полупуть, который проходит через нее и имеет длину $\geq k$, а затем из всех подходящих вершин найти максимальную и удалить ее правым удалением.

Для каждой вершины найдем *height* - максимальное расстояние до вершин в поддеревьях.

Будем для каждой вершины t перебирать предка p и считать, что полупуть проходит через p и не проходит через предков вершины p , т.е. идет от самого дальнего из потомков вершины t до вершины t , затем от вершины t до вершины p , а затем от вершины p до самого дальнего ее потомка в поддереве, в котором нет вершины t .

Сложность работы алгоритма $O(n^2)$, так как для каждой вершины нужно произвести количество операций, равное расстоянию от этой вершины до корня дерева. Если дерево в условие бы было сбалансированным, то сложность работы алгоритма была бы $O(n * \log_2(n))$