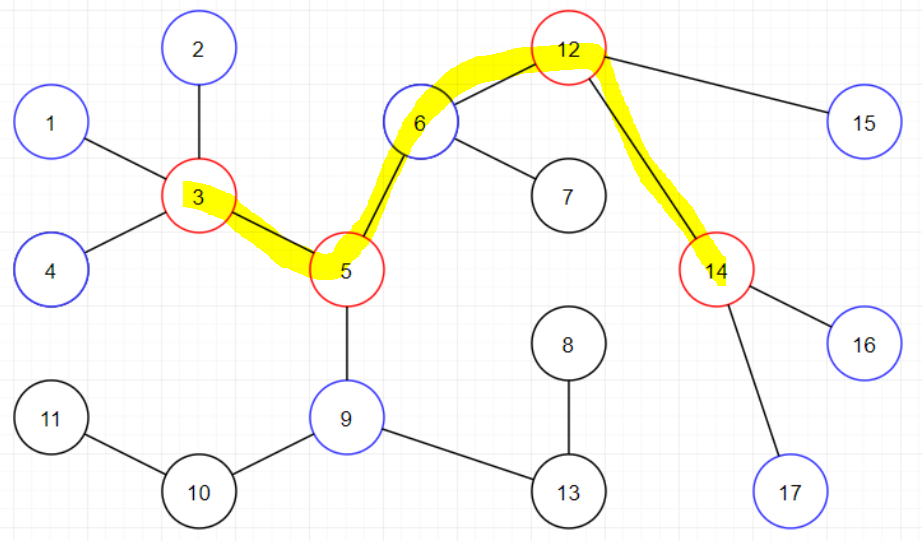
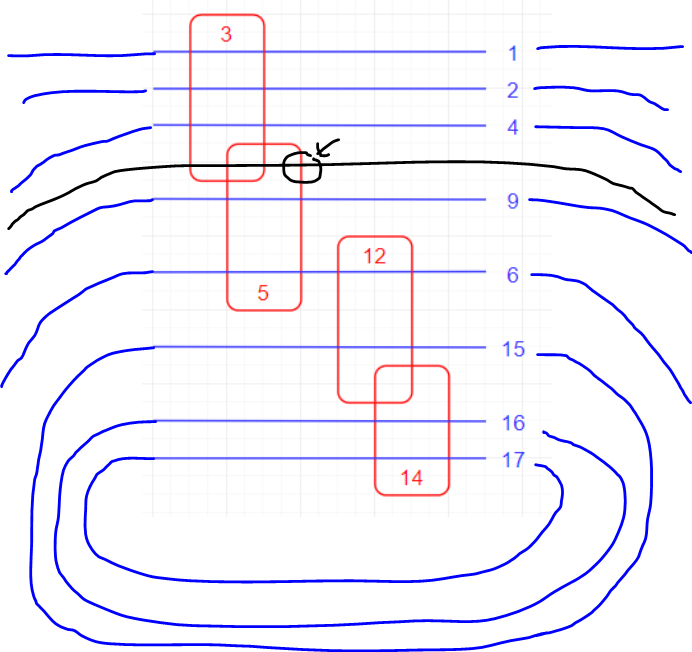
Given a tree with vertices. Convert this tree into loops on a plane. Loops intersect only if an edge exists between them in the tree. The loops cannot self-intersect. A loop includes another loop if they don't intersect and one of them is completely inside another. Find the maximum number of nested loops.

Generate an optimal sequence which is subsequence of a path between some 2 vertices. The neighbours of vertices in the optimal sequence (that is not in the optimal sequence) will be the nested loops.



Black vertices cannot be a part of the nested loops because it must be between 2 blue lines, which means it must intersect a red line, but black and red vertices are not connected in the tree.

observation: 2 red vertices at most 2 edges apart (obviously)

tree dp!!