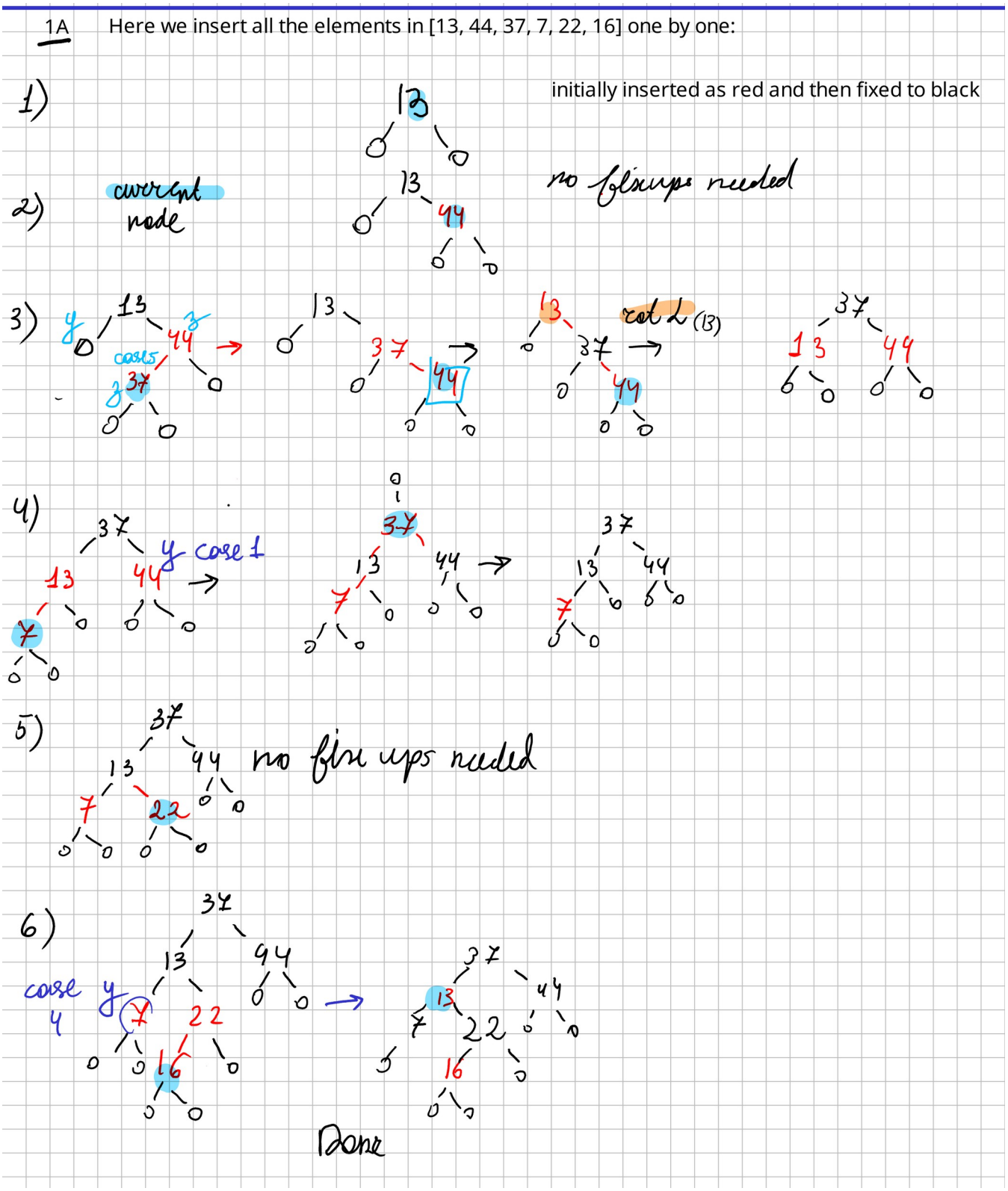
Homework 10 by Timofei Podlorytov

10.1

a)

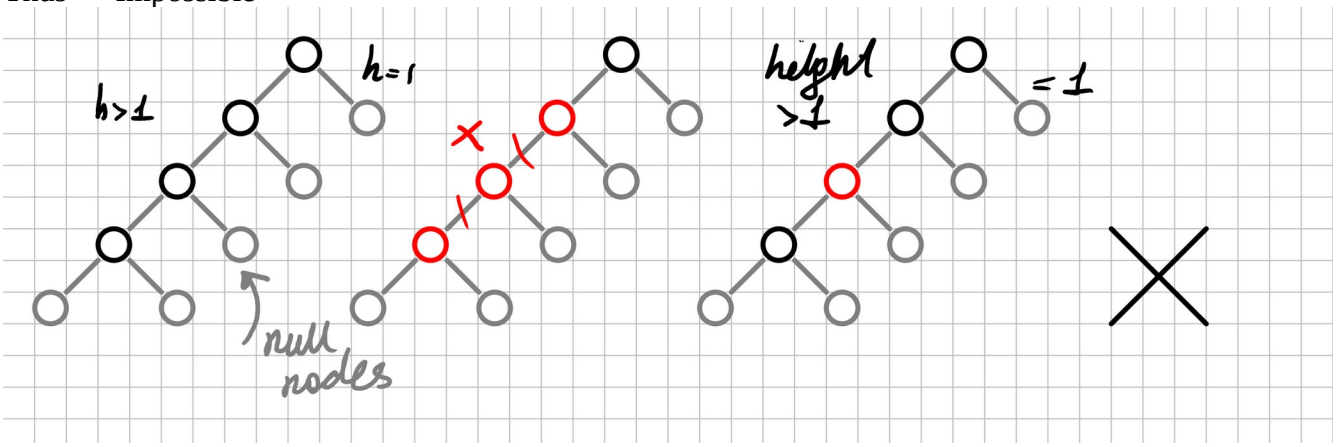
I drew all the insertions step by step. The process is displayed below:



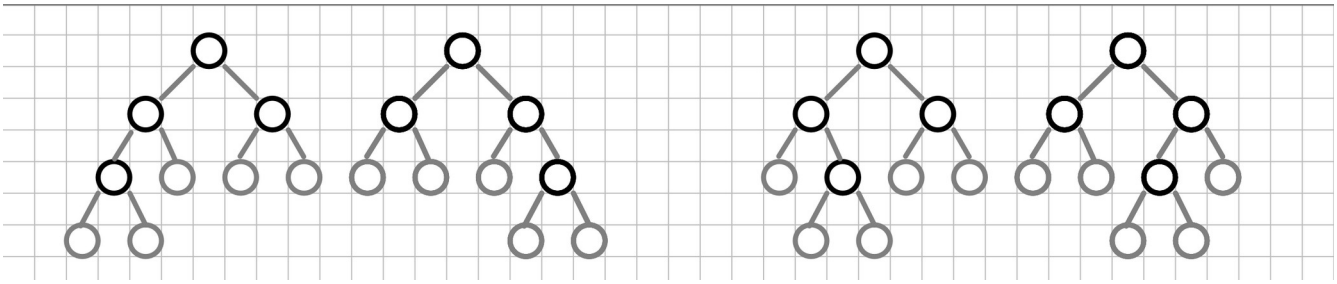
b)
We want all possible variations of Red and black trees for {1,2,3,4}.

Here we first try to narrow down the node positions. We look at them regardless of values inside only thinking about the tree properties and their violations

Case1:
firstly, let's think of possible arrangements regardless of the values there. If we arrange all the elements in one branch we won't have the same black height as on the right it would be 1. While on the left if every node is red, a red node will have a red child → Contradiction. But, if some are black then the height is larger then 1. Contradiction.
Thus → Impossible

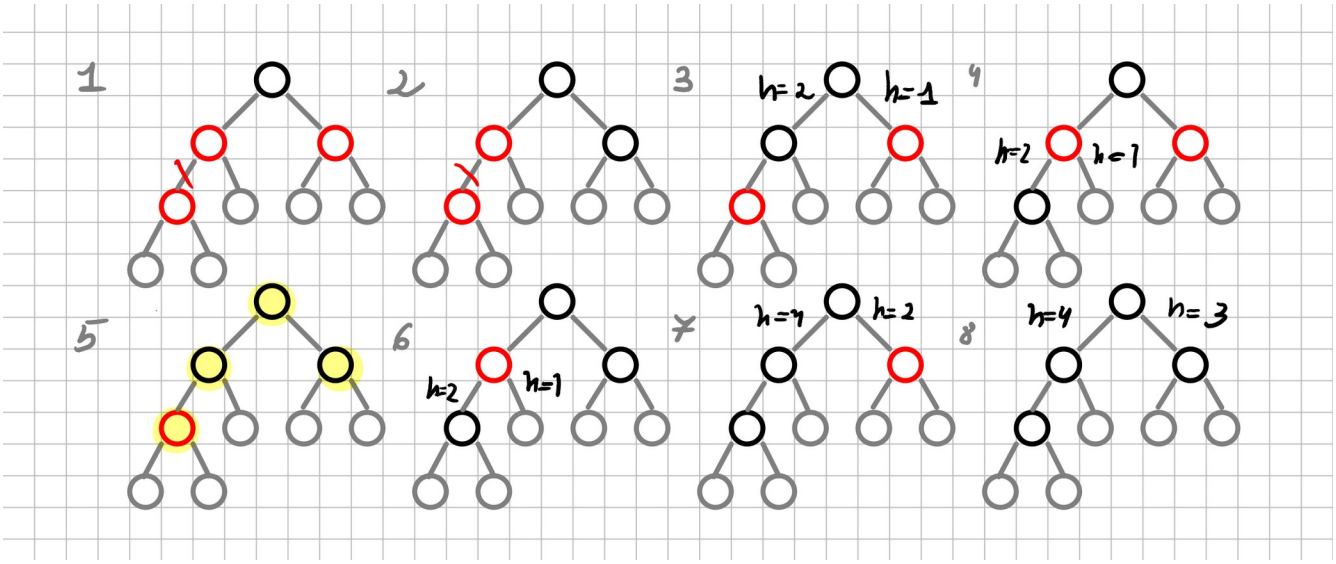


Case2:
We can have 4 arrangements of nodes regardless of color, which are either symmetrical or equal from the point of black height of nodes. So if a pattern follows one of these arrangements we can get 3 more.

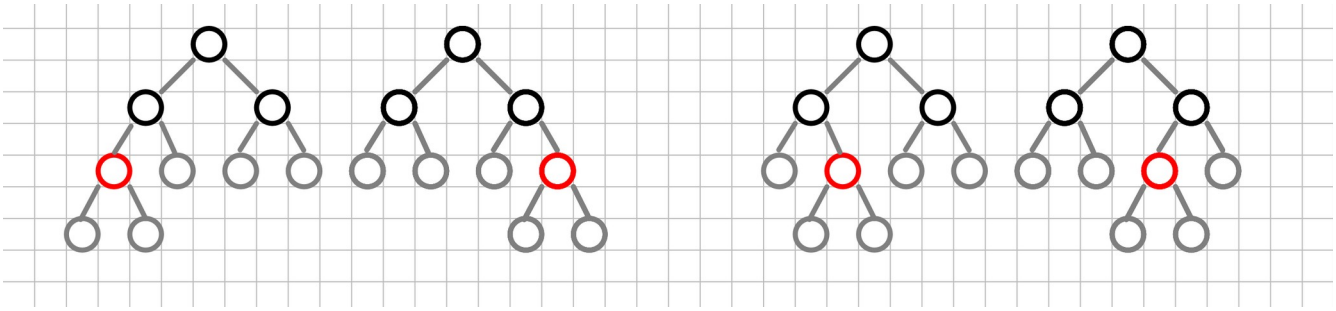


→see next page

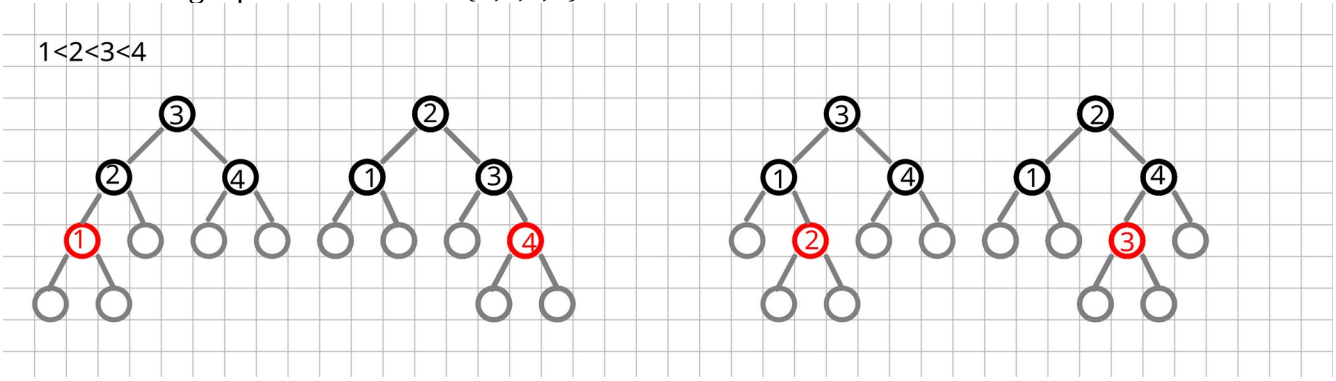
Root is always black. For the other 3 nodes they can be black or red giving us 8 possible cases.
For 1 and 2 the red node has a red child and for the other 5 cases we have mismatch between black height. The only case that can work according to red and black tree conditions is **case 5.**



From case 5 we derive its four variations which are equal to it.



What we can do now is fill the nodes with data. First, lets find the leftmost node-the smallest value 1. Then we can find it's successor and put value 2 there. Similarly we can decide on where to place 3 and 4. The resulting 4 possible trees for {1,2,3,4} are drawn below.



Now we know the answer and we know there are no more solutions as those are all the ways to arrange 4 nodes in a binary tree without disrupting it and breaking red and black tree conditions.

10.2

I implemented the tree in 2 files. One for declaration of the class and the other for defining the methods of it.

There's also a run.cpp file where all the methods are tested and shown to be working properly. I added 2 printing methods in public methods of the class as I thought that they would be useful for checking the execution. Some private and protected methods were also added by me when needed.