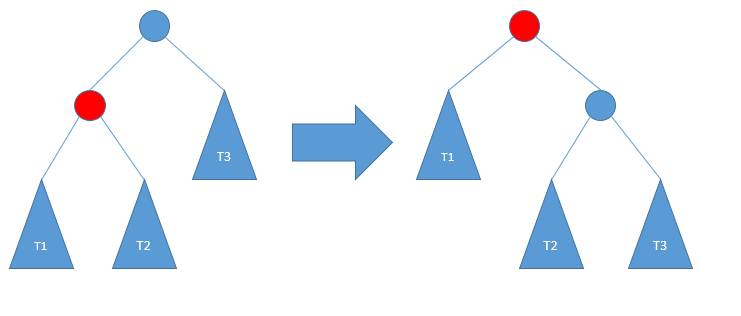
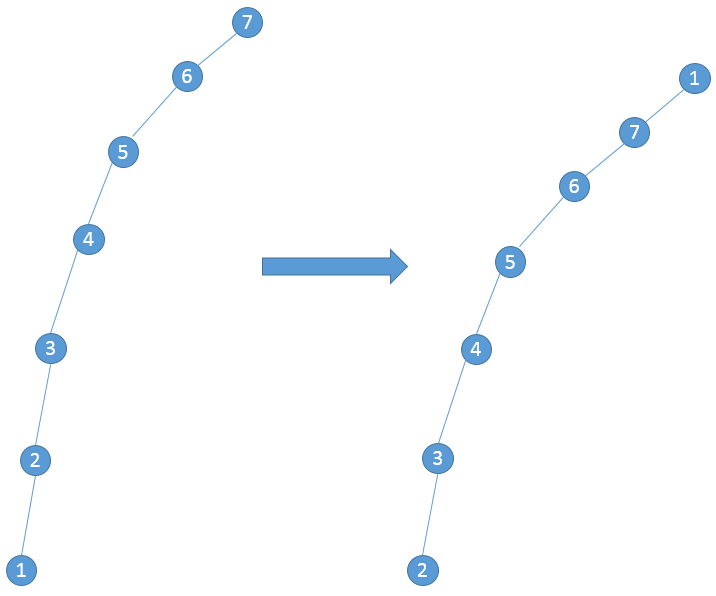
Алгоритмы и структуры данных. Домашняя работа. Неделя 2

Автор: Петровский Александр M3139

Задание 1

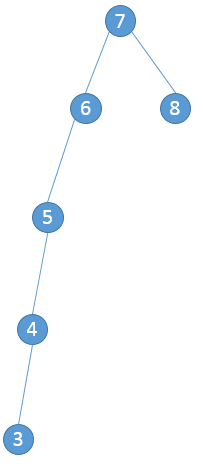
Переходы при таком переходе представлены на рисунке. Решив третью задачу и получив бамбук, можно задумываться, а не он ли помощь к этой задаче. Пусть у нас есть бамбук, который вытянут целиком влево (как в третьей задаче, только отсортируем: в корне максимальный, а в листе минимальный и тд), тогда применив find к последнему n-ому элементу, тогда если он будет делать простые повороты zig, то

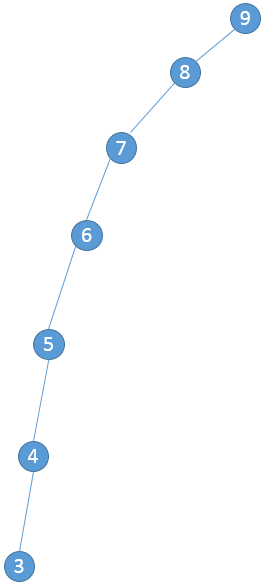


 получим такое дерево. Очевидно, что поднимая так все вершины снизу вверх мы снова получим исходное дерево. То есть мы будем выполнять каждую операцию за O(n).

Задание 2

Задание 3



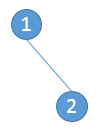
Если добавить вершину 90, то вершина 80 попадет под splay, от чего на время окажется корнем, после добавляется 90, как родитель текущего корня, как результат получаем бамбук.

Если вставим 55, то у дерева будут две одинаковые ветки, а высота 4, если будем добавлять больше/меньше, то за счет zig-zig могли бы получить и меньше, но если нарисовать, например 35, то из-за двойных zig-zig мы не получем высоту лучше 4 (но все ту же высоту 4).

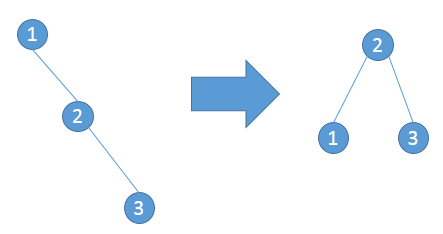
Задание 4

После первого шага добавиться одна единственная вершина, которая и станет корнем.

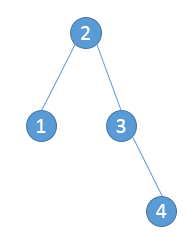


Вторая вершина пойдет в правое поддерево, так как она больше первой.

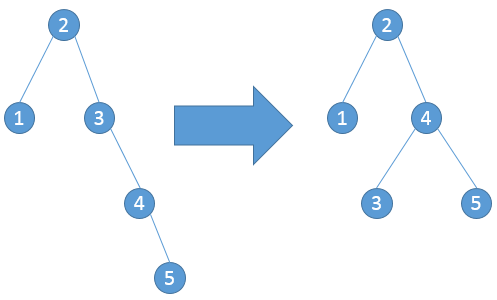
После вставки третьей вершины нужно сделать сплит, который устранит два последовательных правых «горизонтальных» ребер.



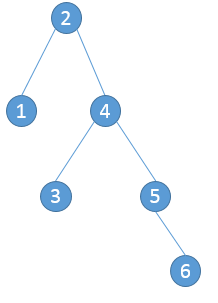
Четвертая вершина займет свое место без проблем.



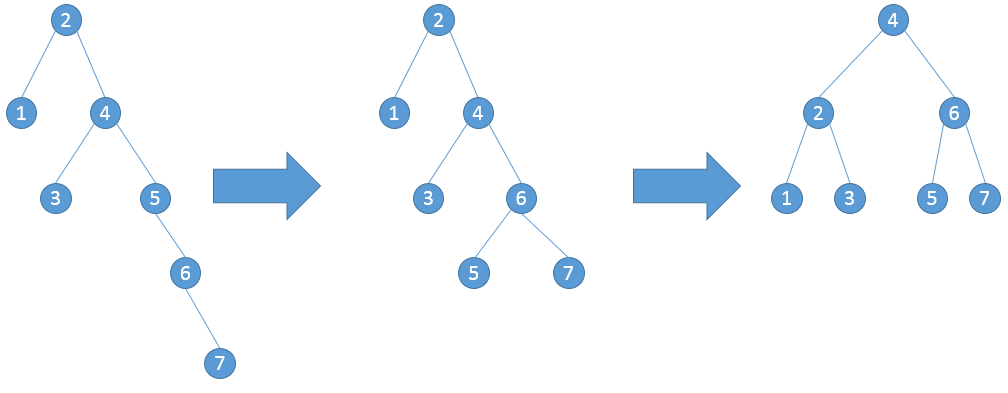
После вставки пятой вершины снова делаем Split.



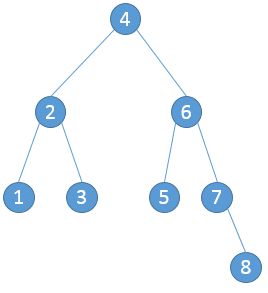
Вставляем спокойно шестую вершину.



После вставки 7-ое вершины дважды получаются правильные горизонтальные связи, поэтому применим дважды сплит.



Добавление 8-ой вершины.



Добавление 9-ой вершины.

