See visualizations:

<https://www.cs.usfca.edu/~galles/visualization/RedBlack.html>

<https://www.cs.usfca.edu/~galles/visualization/BST.html>

A **red–black tree** is a kind of self-balancing binary search tree.

A red-black tree with ***n*** internal nodes has height at most 2**log(*n*+1)**.

The balancing of the tree is not perfect, but it is good enough to allow it to guarantee searching in [O(log *n*)](https://en.wikipedia.org/wiki/Big-O_notation) time, where *n* is the total number of elements in the tree.

During insert and delete operations, nodes may be rotated to maintain tree balance.

Because std::map is implemented as a binary search tree, it sorts every element we put into it! Sometimes, in code we want to have mapping between key and value, but we may not need the sorting part. In that case, consider other containers (like std::unordered\_map).