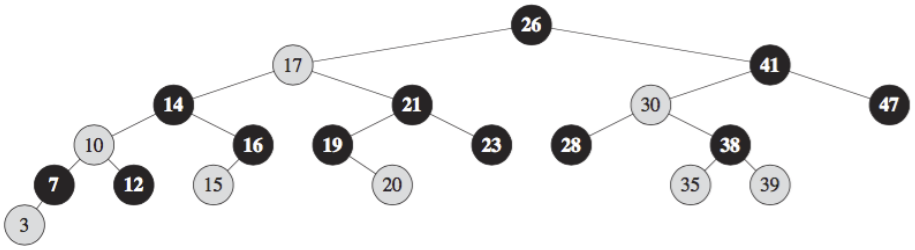
Werkcollege 10

You are allowed to answer in Dutch. Whenever an algorithm is required, it can be given in pseudocode or plain English (or Dutch), and its running time and correctness must always be justified (even informally, but in a clear way!).

10.1. Apply the insertion procedure for ordinary binary trees on the red-black tree below (grey nodes stand for red-colored ones), with key 36, and draw the result. If the inserted node is colored red, is the resulting tree a red-black tree? What if it is colored black? If not, specify which properties are violated.



10.2. What is the largest possible number of internal nodes in a red-black tree with black-height k? What is the smallest possible number?

10.3. Draw the red-black trees that result after successively inserting the keys

```
32, 27, 20, 15, 19, 33
```

into an initially empty red-black tree. Delete 19 and then 32, and draw the resulting trees after each deletion.

10.4. Suggest how to implement insertion in a red-black tree if nodes do not include parent pointers. The running time must still be $\mathcal{O}(\log n)$.

10.5. Show that any arbitrary n-node binary search tree can be transformed into any other arbitrary n-node binary search tree on the same set of nodes, using $\mathcal{O}(n)$ rotations.

Hints: show that you need $\mathcal{O}(n)$ rotations to transform any tree into one where each node either has no children or it has a right child.