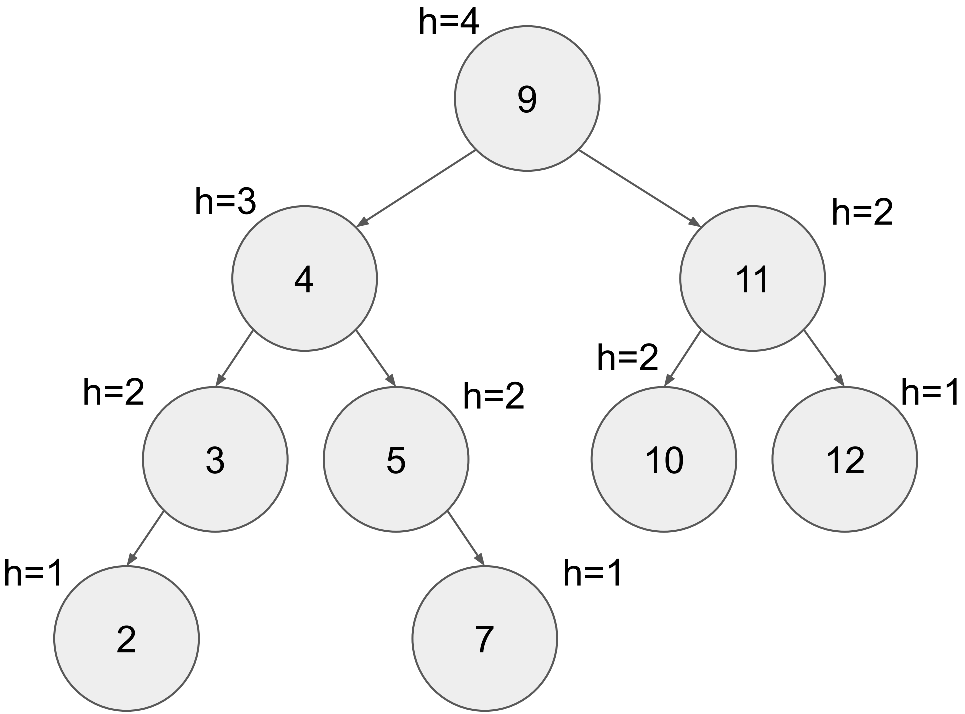
List of operations

- Remove 10
- Insert 13
- Insert 6
- Remove 9
- Insert 14
- Remove 11

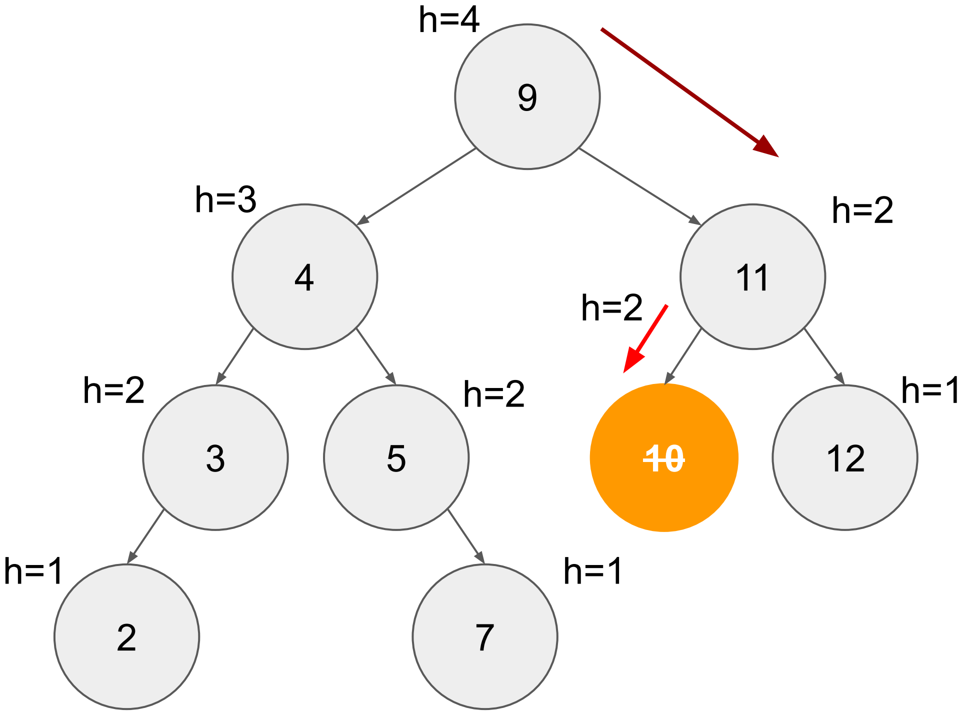


- Remove 10

→ 10 ≥ 9, move right

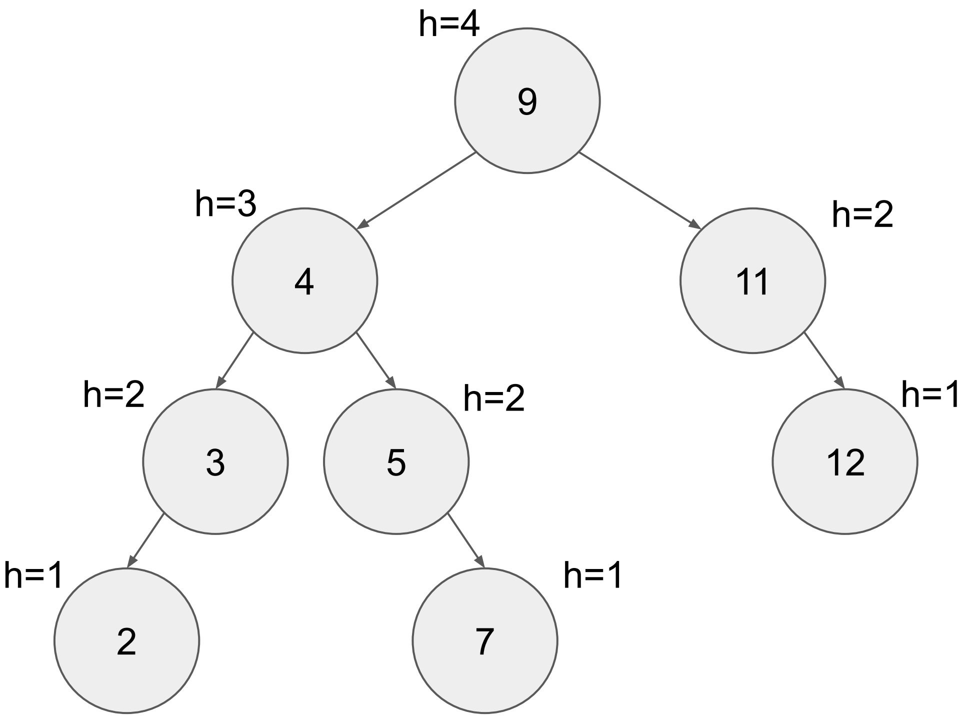
→ 10 < 11, move left

→ Insert 10



List of operations

- ~~Remove 10~~
- Insert 13
- Insert 6
- Remove 9
- Insert 14
- Remove 11



- Insert 13

→ 13 ≥ 9, move right

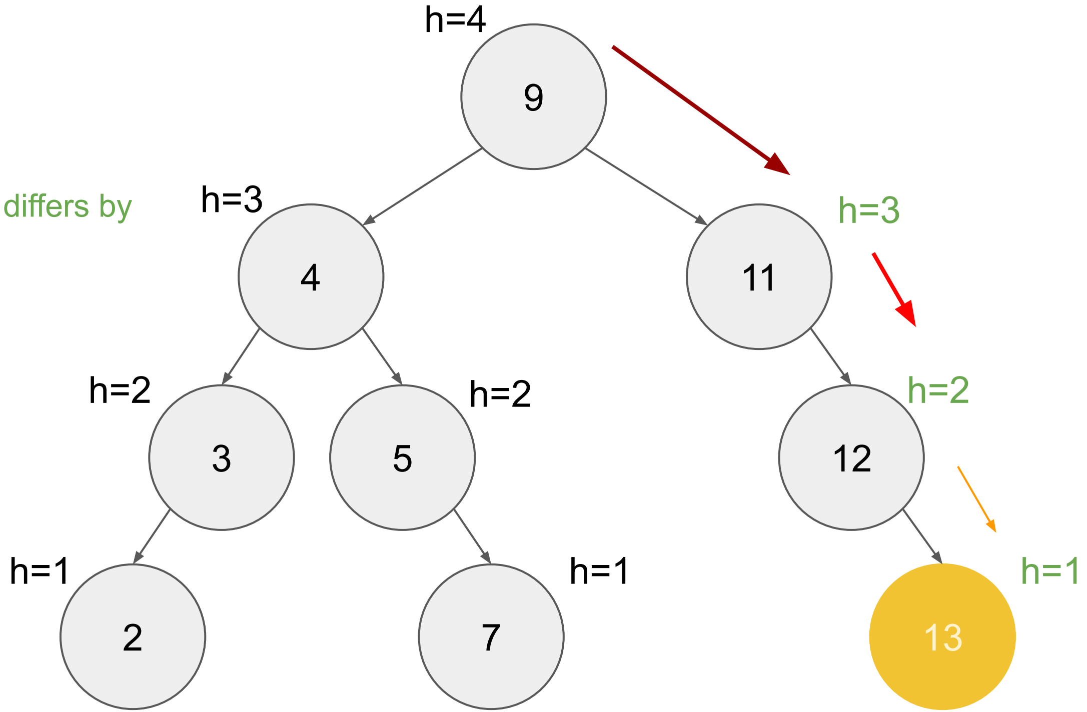
→ 13 ≥ 11, move right

→ 13 ≥ 12, move right

→ insert 13

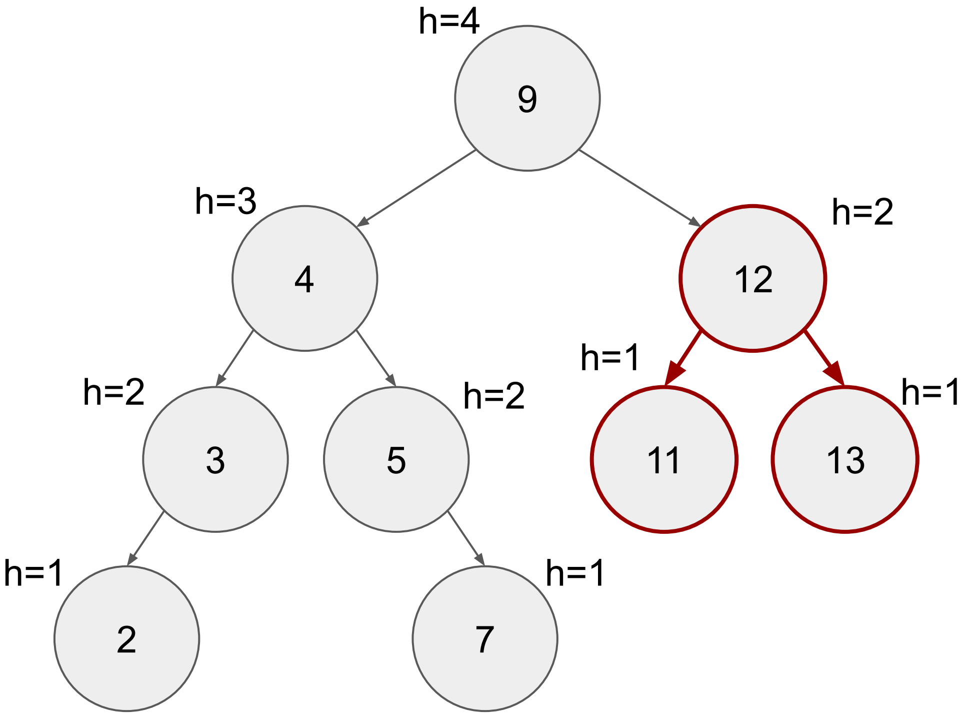
→ adjust height

Height of left/right tree for 11 differs by more than 1.



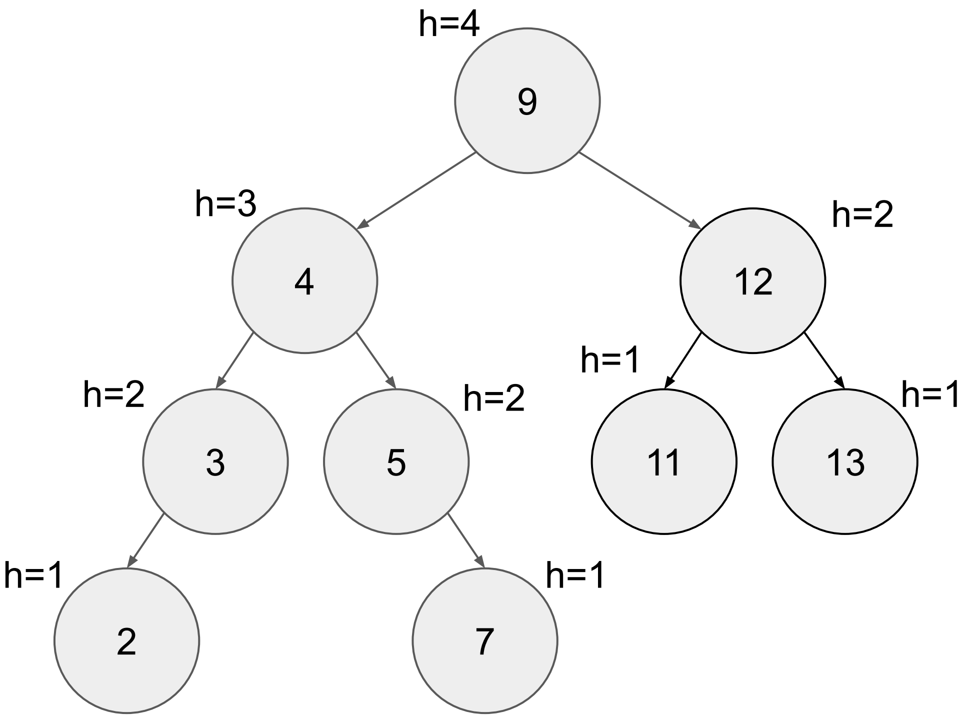
- Insert 13

→ Single Rotate left



List of operations

- ~~Remove 10~~
- ~~Insert 13~~
- Insert 6
- Remove 9
- Insert 14
- Remove 11



- Insert 6

→ 6 < 9, move left

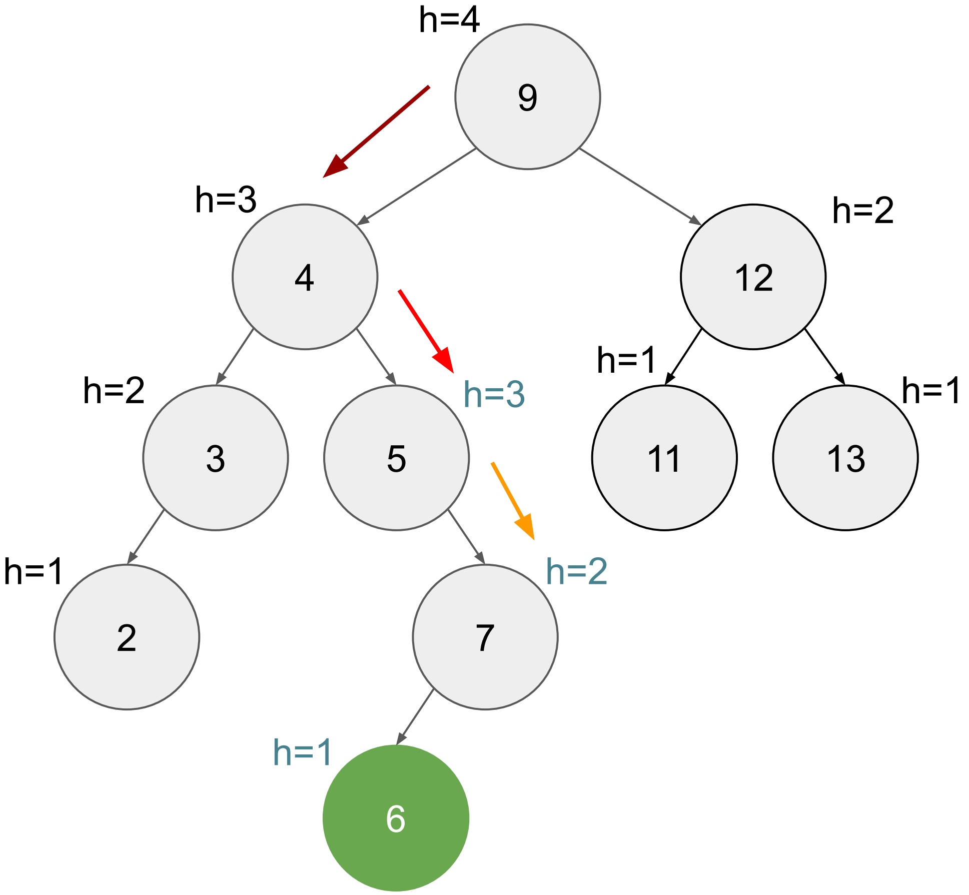
→ 6 ≥ 4, move right

→ 6 ≥ 5, move right

→ 6 < 7, move left

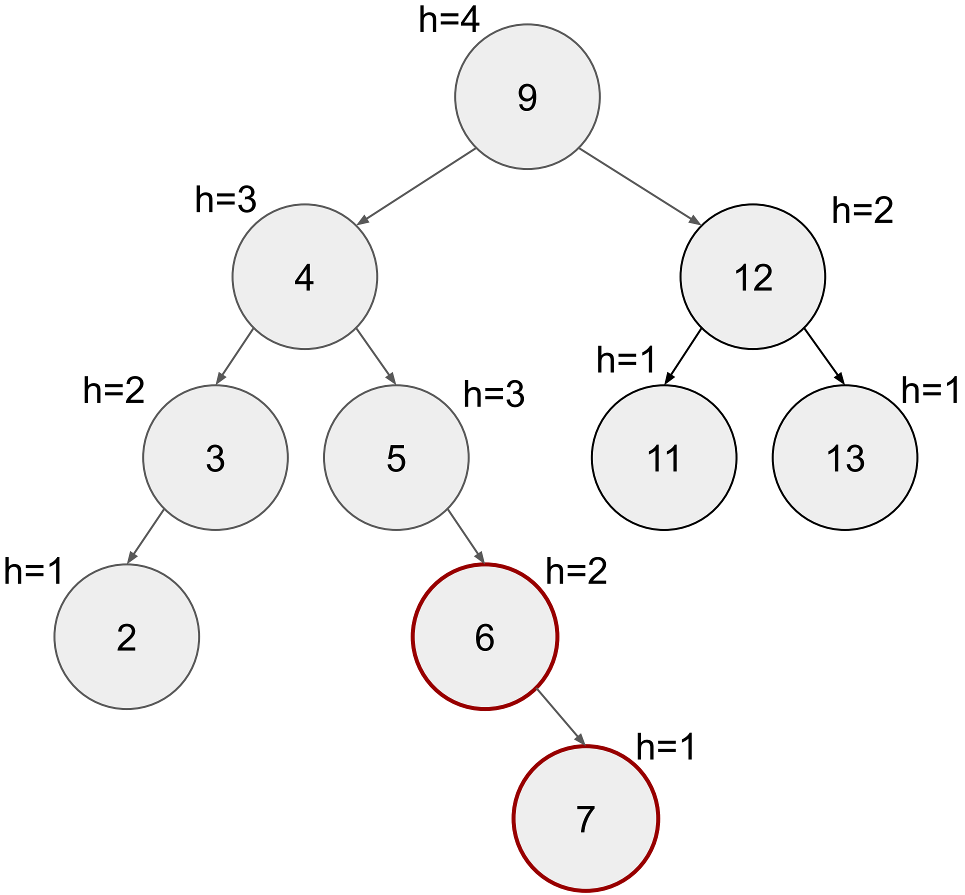
→ insert 6

→ adjust height



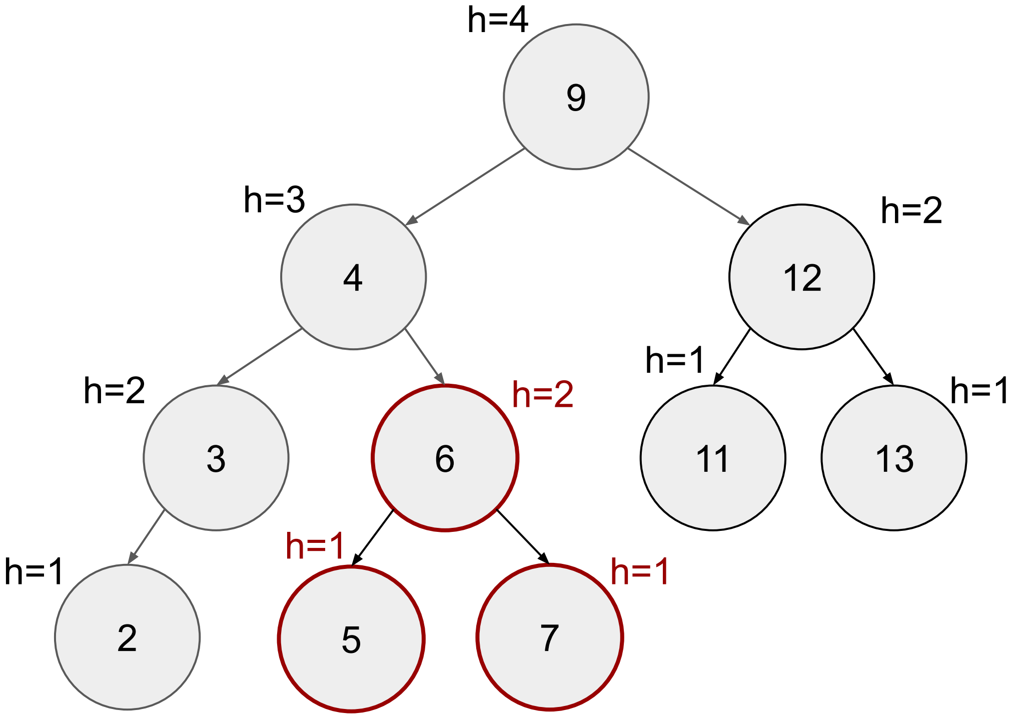
- Insert 6

→ Double rotate left, part 1



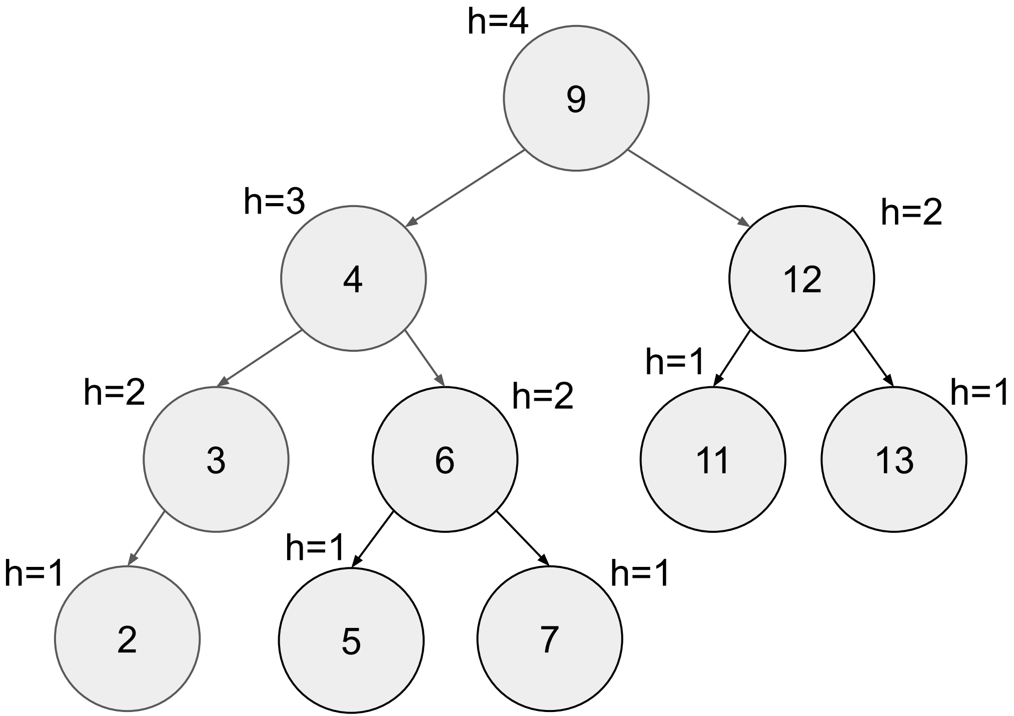
- Insert 6

→ Double rotate left, part 2



List of operations

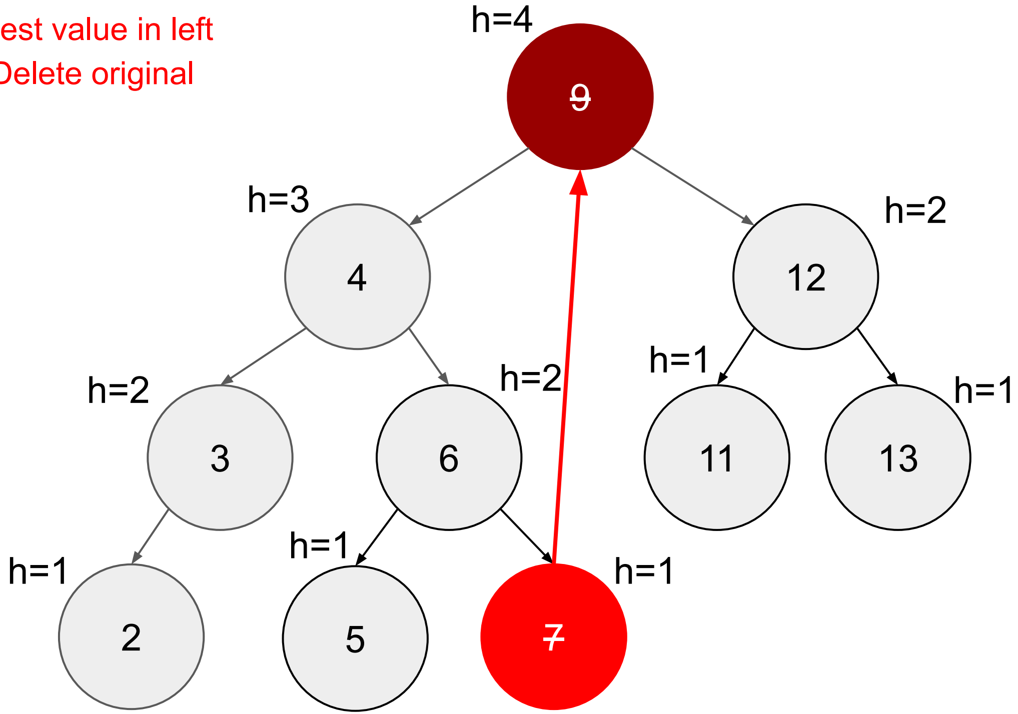
- ~~Remove 10~~
- ~~Insert 13~~
- ~~Insert 6~~
- Remove 9
- Insert 14
- Remove 11



- Remove 9

→ Found 9, delete

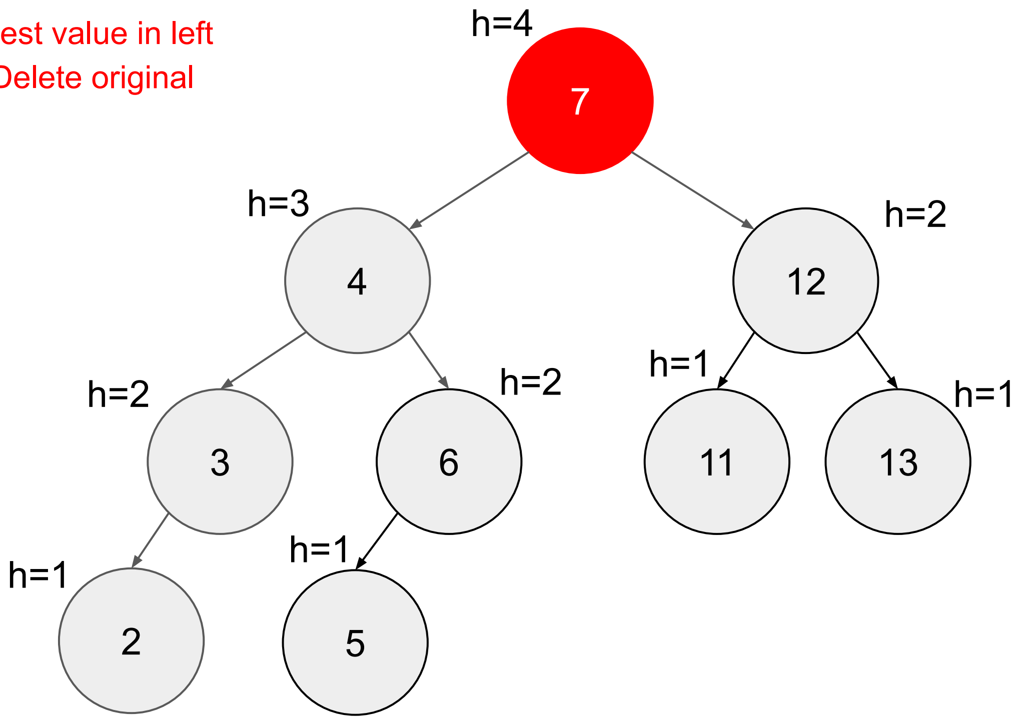
→ Two children: Find largest value in left subtree and copy to top. Delete original



- Remove 9

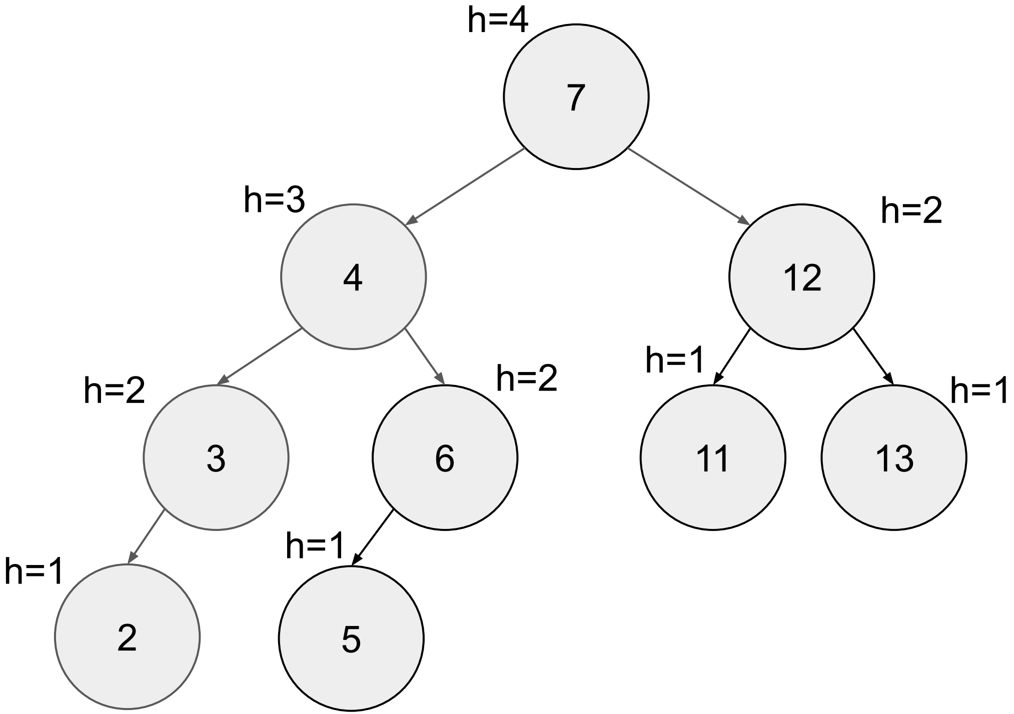
→ Found 9, delete

→ Two children: Find largest value in left subtree and copy to top. Delete original



List of operations

- ~~Remove 10~~
- ~~Insert 13~~
- ~~Insert 6~~
- ~~Remove 9~~
- Insert 14
- Remove 11



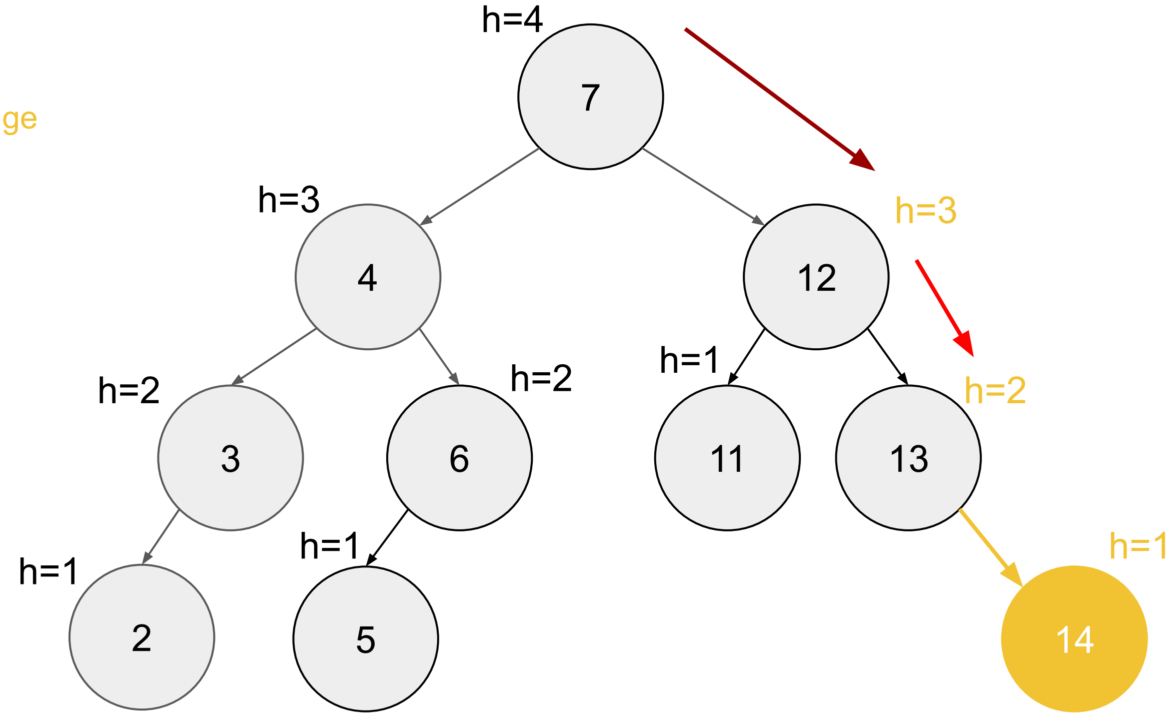
- Insert 14

→ 14 ≥ 7, move right

→ 14 ≥ 12, move right

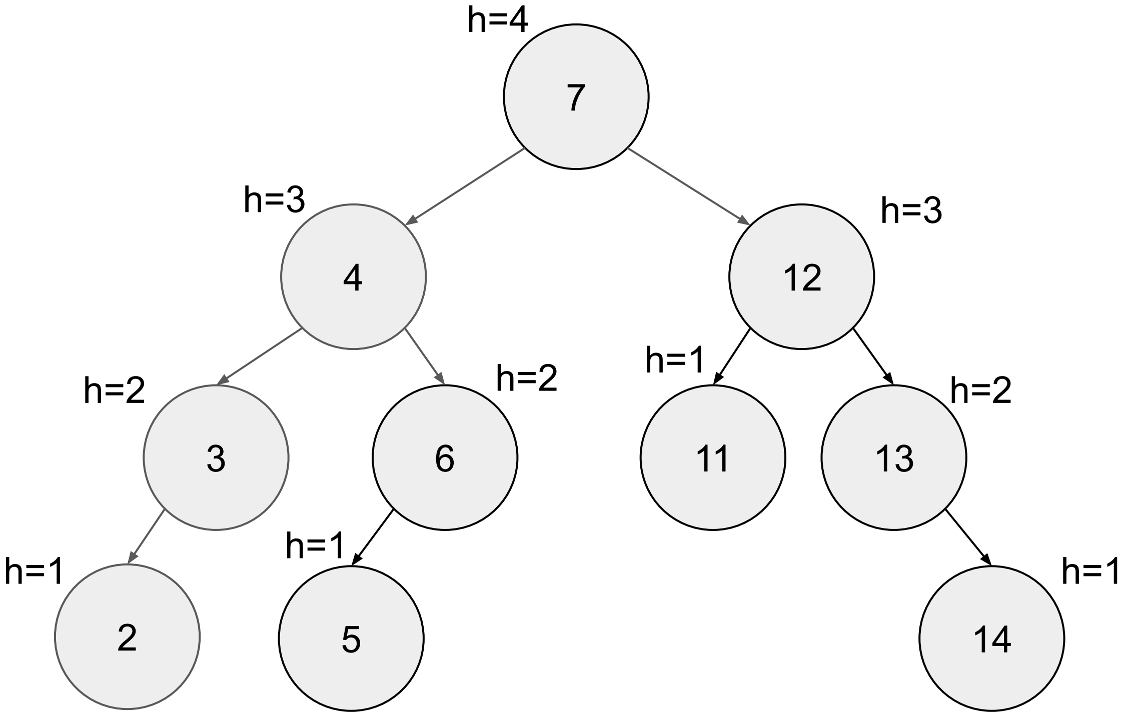
→ 14 ≥ 13, move right

→ Insert 14. Height change



List of operations

- ~~Remove 10~~
- ~~Insert 13~~
- ~~Insert 6~~
- ~~Remove 9~~
- ~~Insert 14~~
- Remove 11

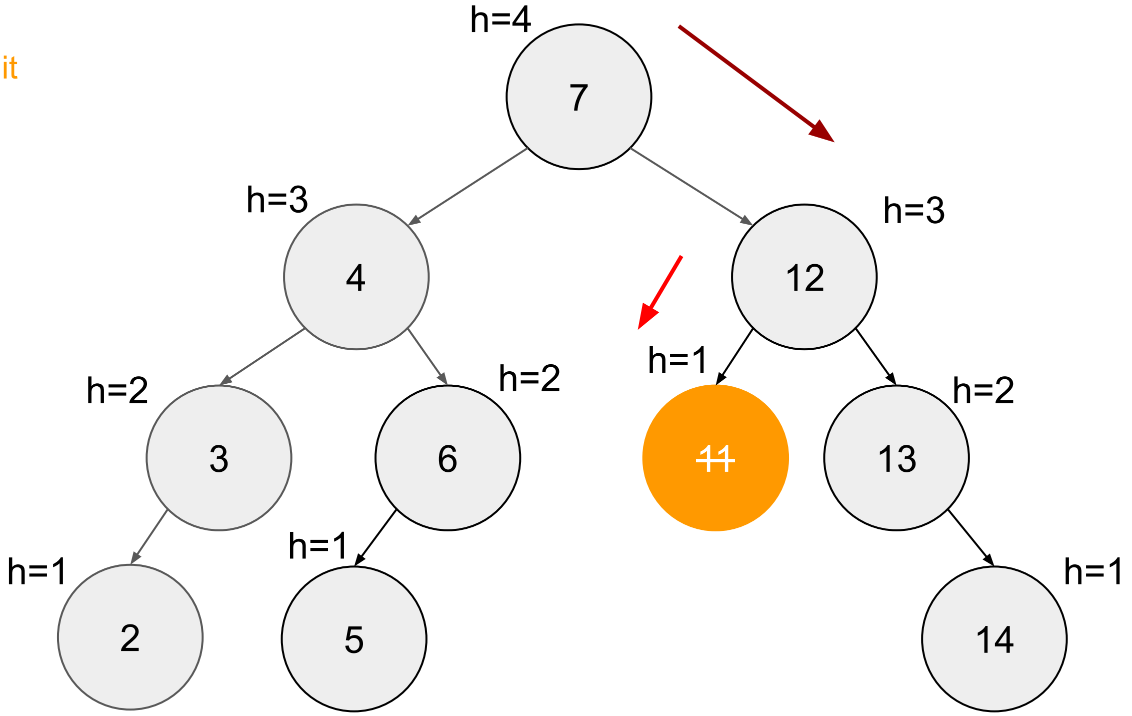


- Remove 11

→ 11 ≥ 7, move right

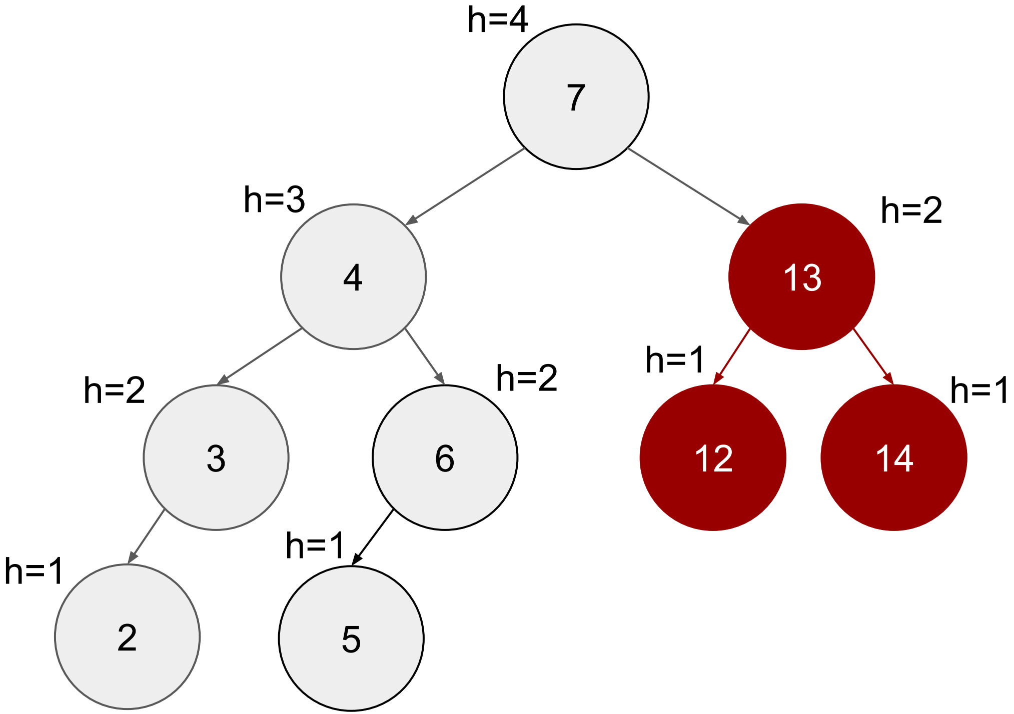
→ 11 < 12, move left

→ 11 == 11!!!!!! Get rid of it



List of operations

- ~~Remove 10~~
- ~~Insert 13~~
- ~~Insert 6~~
- ~~Remove 9~~
- ~~Insert 14~~
- ~~Remove 11~~


h=4
7
h=3
4
h=2
13
h=2
3
h=2
6
h=1
12
h=1
14
h=1
2
h=1
5