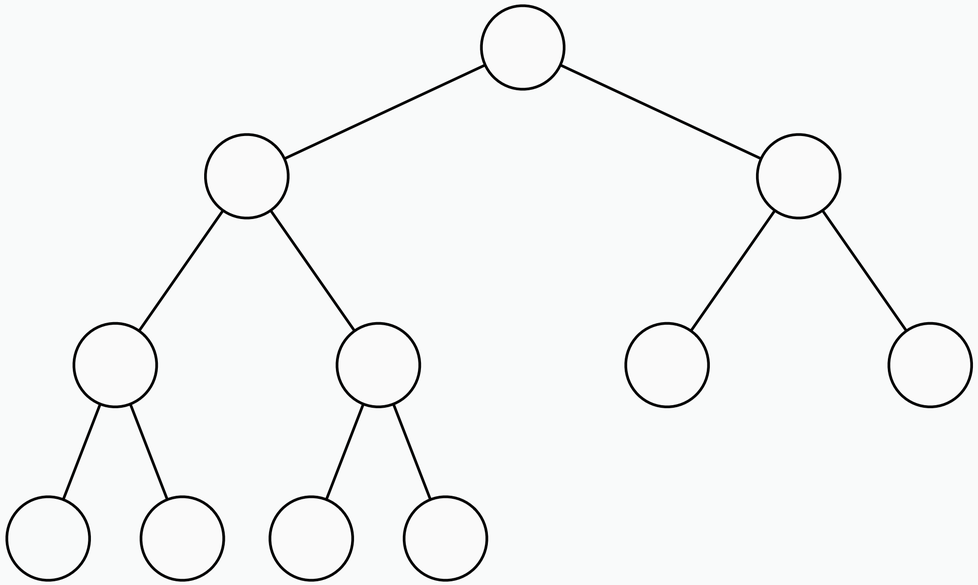


A is a widely used data structure.

Heapsort algorithm (Williams 1964)

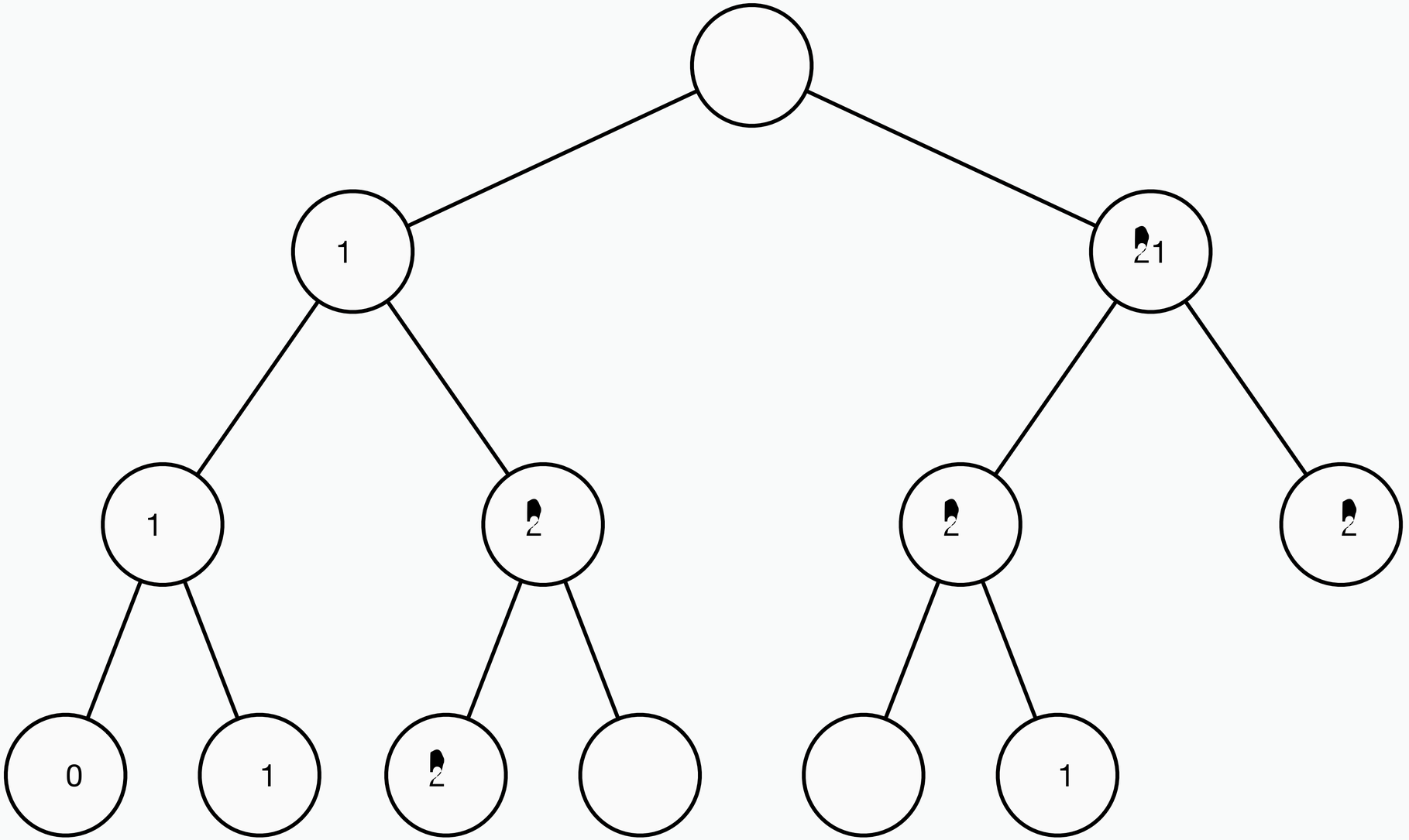
Graph algorithms (. ., shortest path, spanning tree)

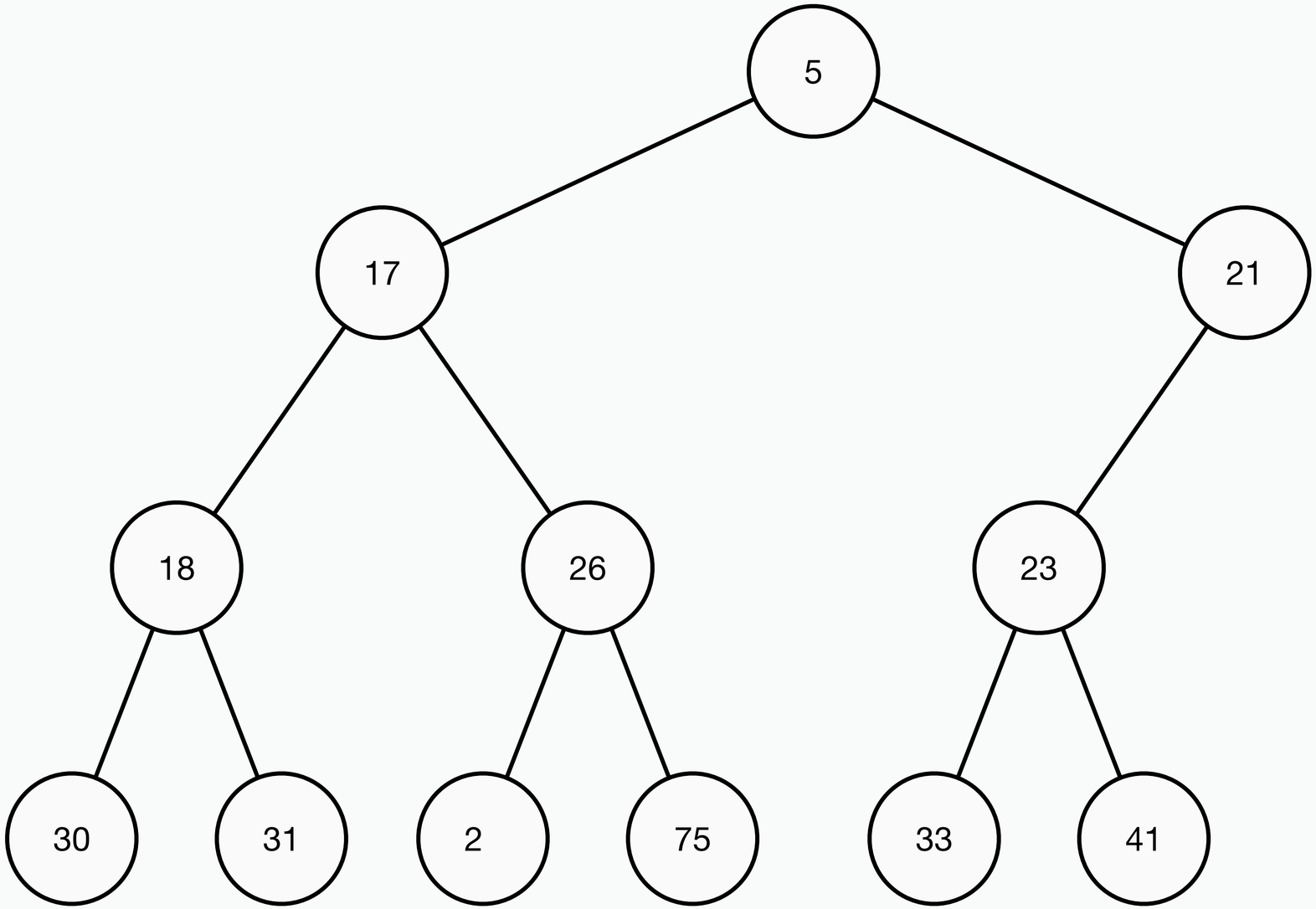
Priority queue (which itself has abundant applications)

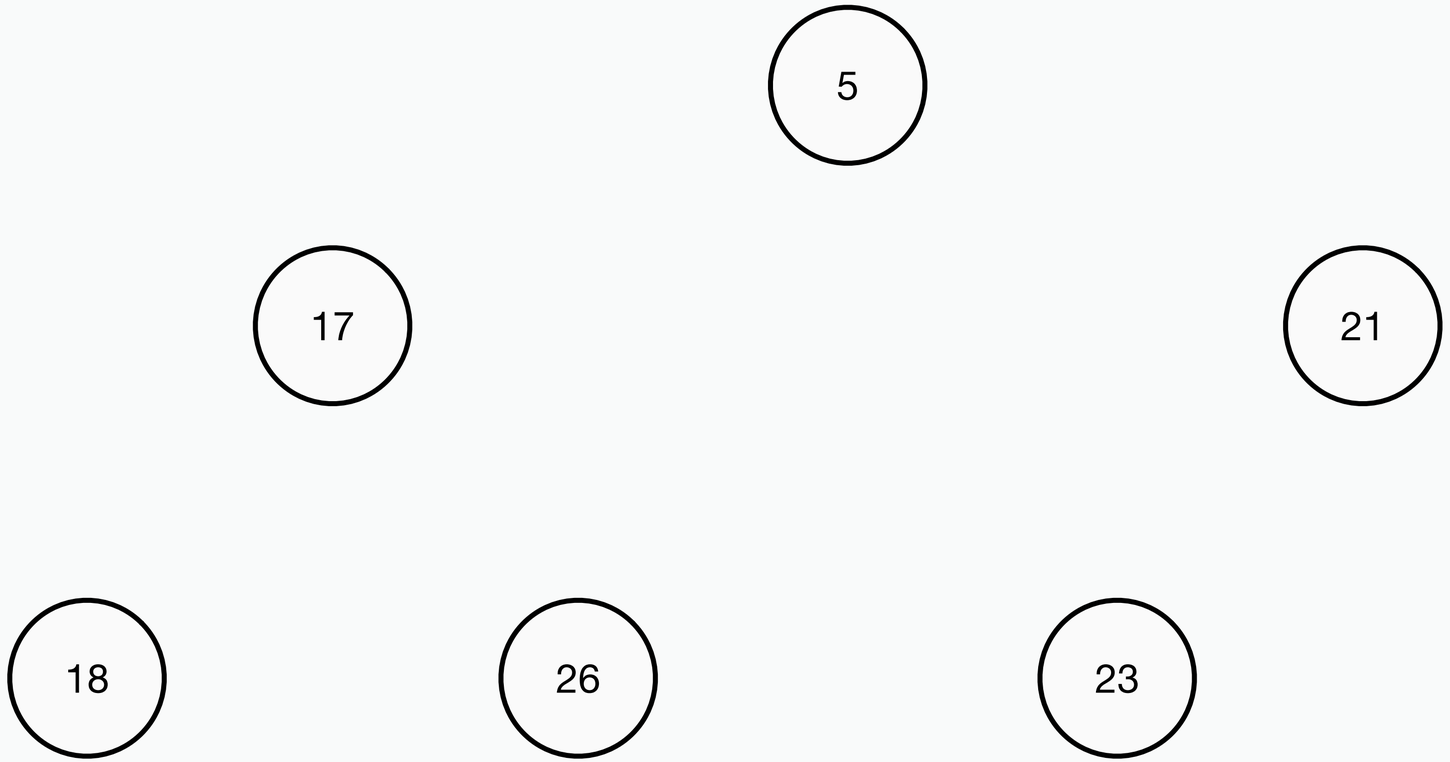
It is important to keep in mind the distinction between a binary heap and a binary search tree (BST). They are the same thing.

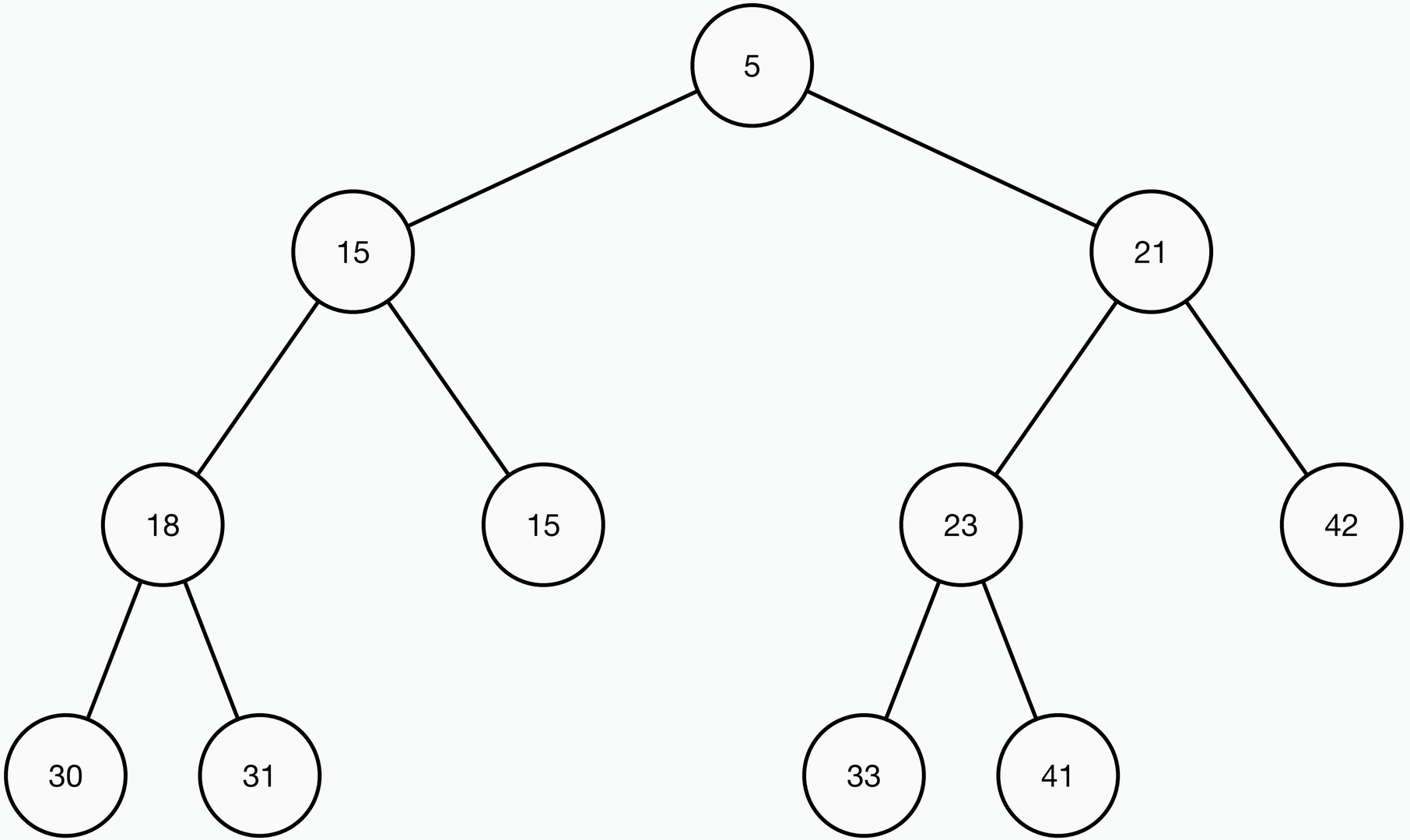
A binary search tree requires that values of a left subtree must be ordered with respect to the right subtree. This is the case with binary heaps. The heap order property is a little more relaxed.

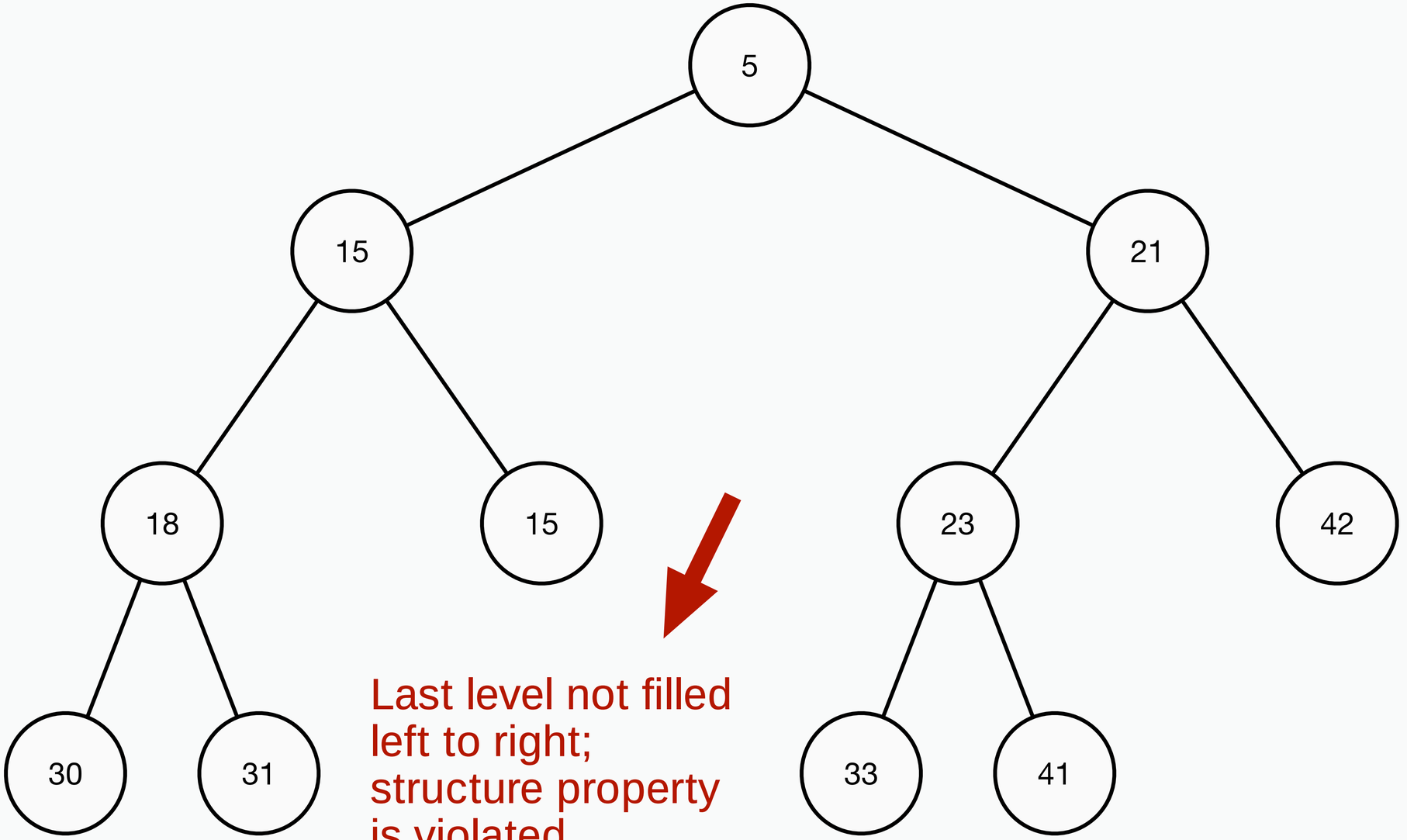
Binary heap binary search tree

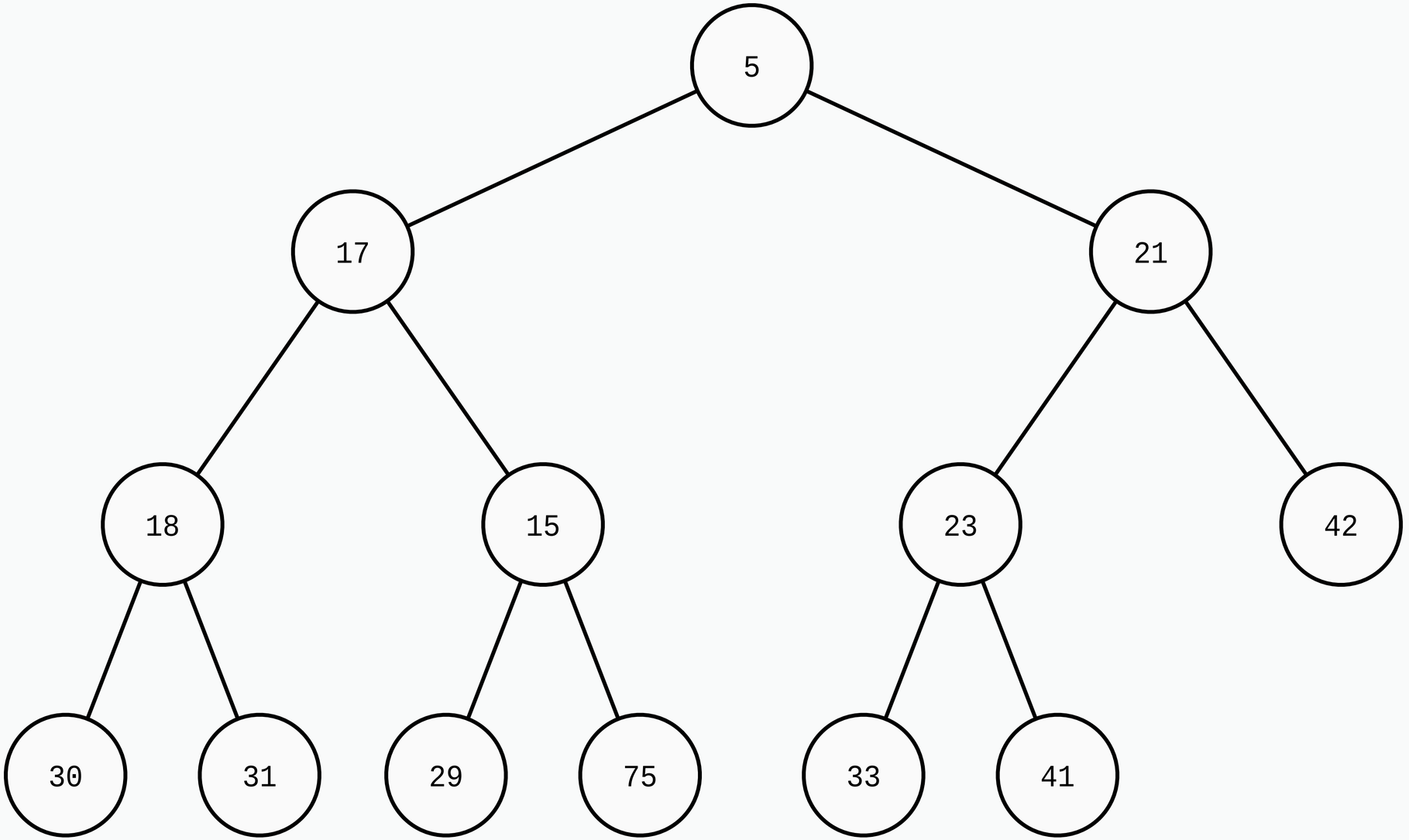

1
21
1
2
2
2
0
1
2
1

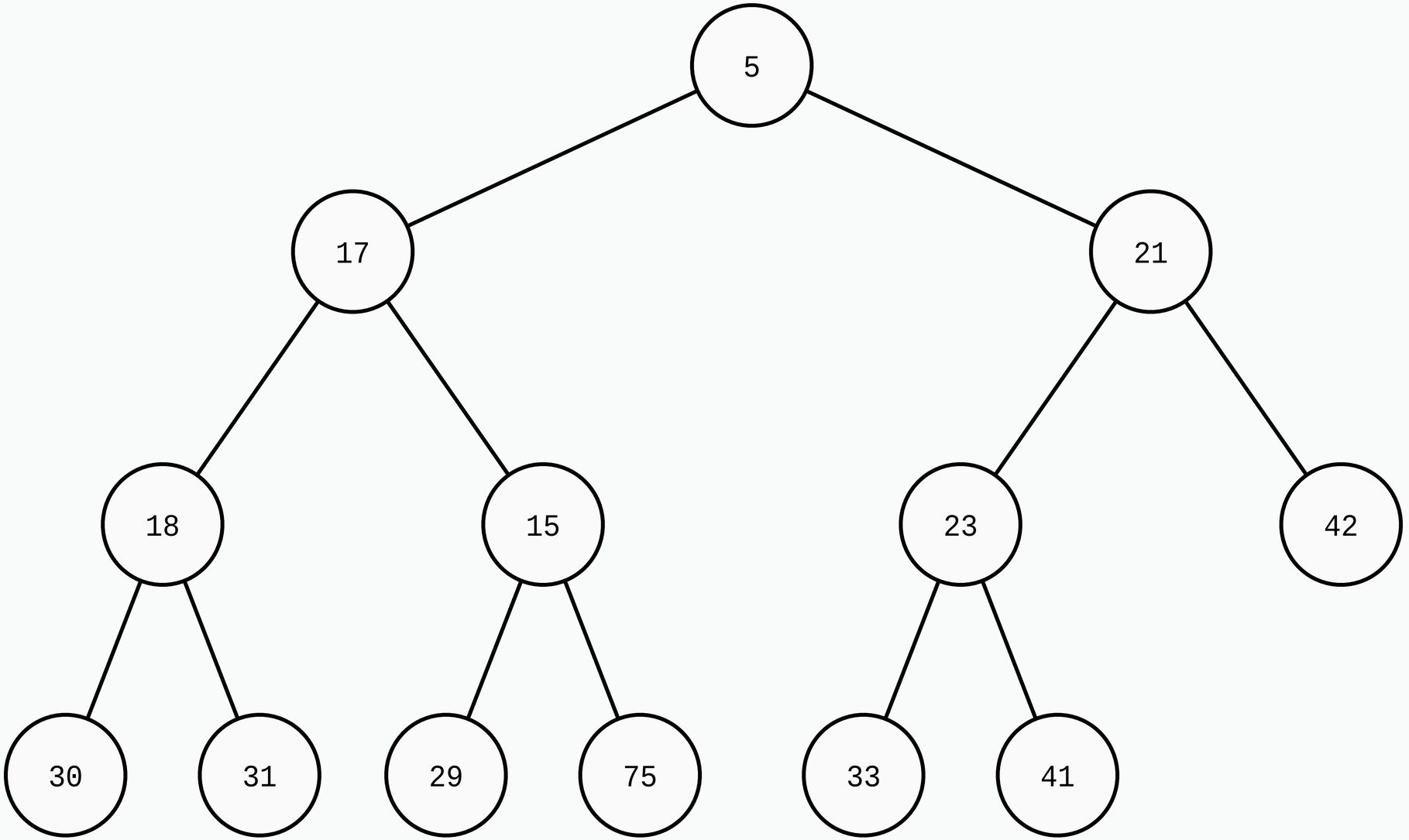

5
17
21
18
26
23
30
31
2
75
33
41

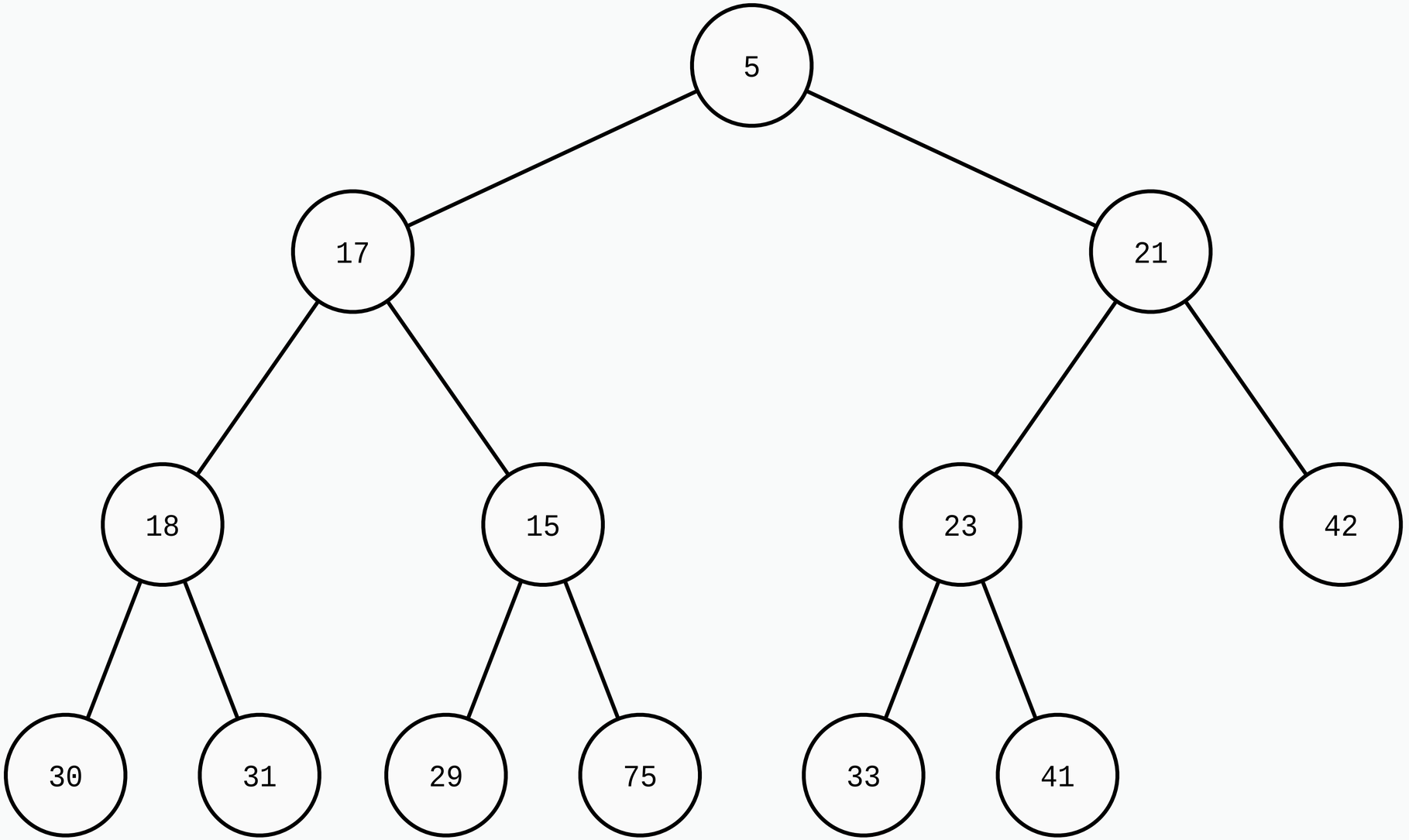

5
17
21
18
26
23

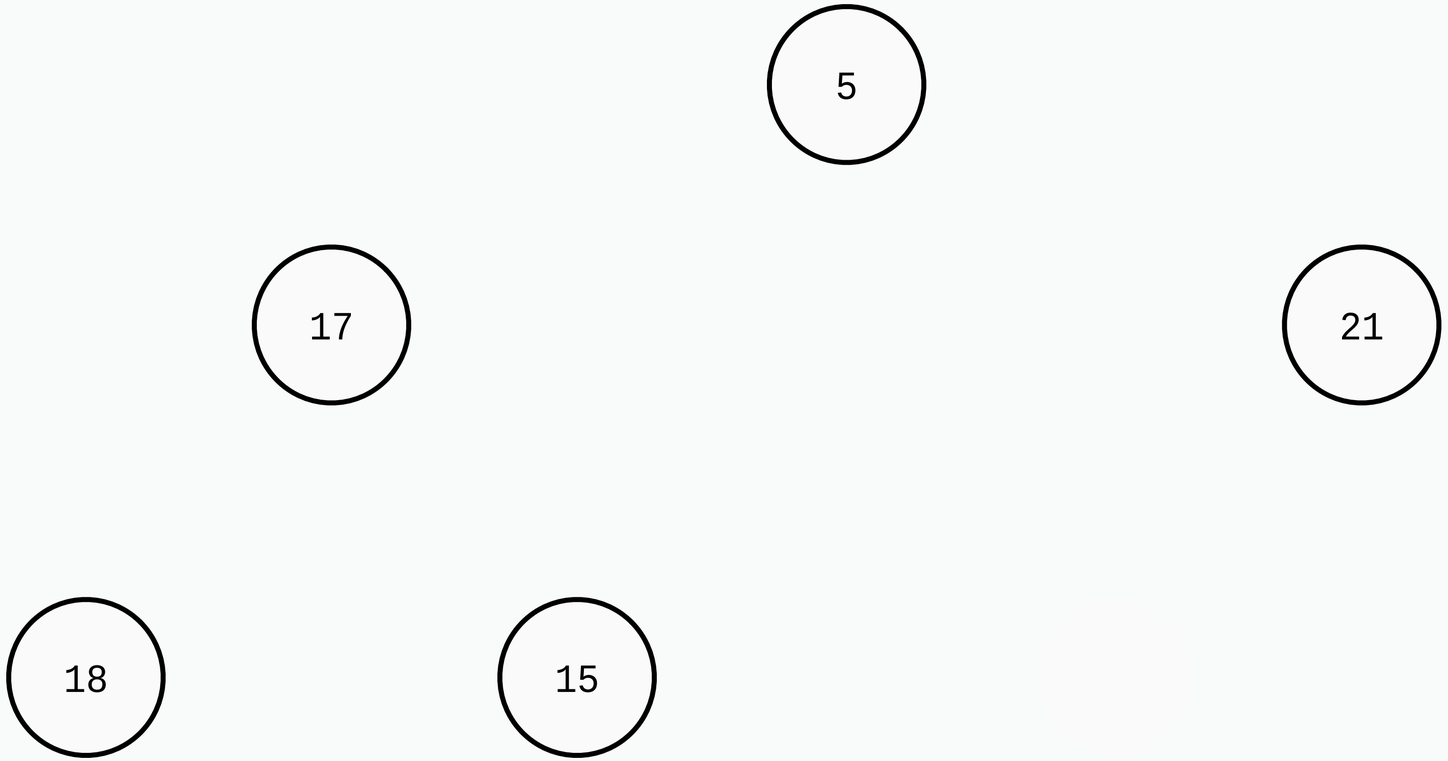

5
15
21
18
15
23
42
30
31
33
41


5
15
21
18
15
23
42
30
31
33
41
Last level not filled
left to right;
structure property
is violated.


5
17
21
18
15
23
42
30
31
29
75
33
41


5
17
21
18
15
23
42
30
31
29
75
33
41

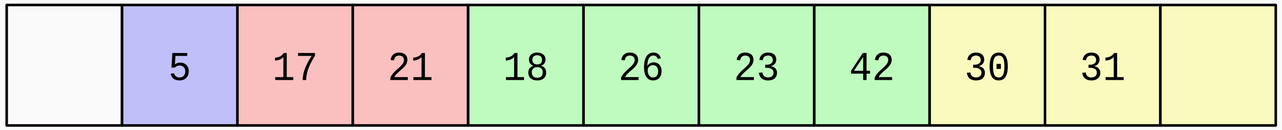

5
17
21
18
15
23
42
30
31
29
75
33
41


5
17
21
18
15

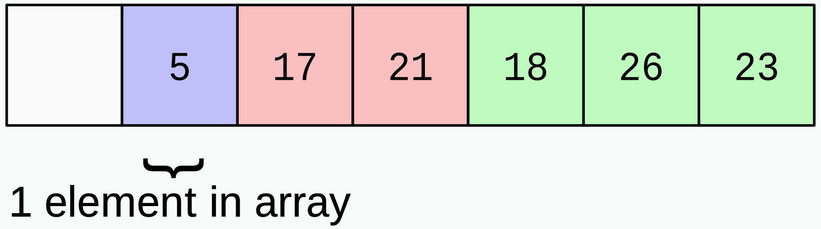
Recall that the of a tree, , is the length of the longest path from the root node to a leaf node.

Recall also that the structure property of a binary heap requires that the tree be . A complete binary tree must have between 2 and 2^{+1} -1 nodes.

Given a tree of nodes, its height is $\lfloor \quad _2 \rfloor$.

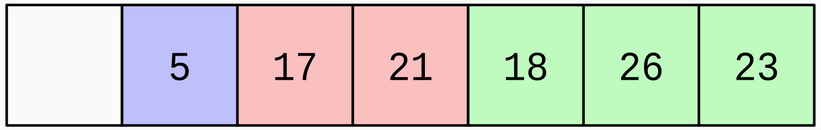

5
17
21
18
26
23
42
30
31

Because a binary heap has a highly regular structure, we can represent it with an .



2^0 = 1 node in this level

Because a binary heap has a highly regular structure, we can represent it with an .

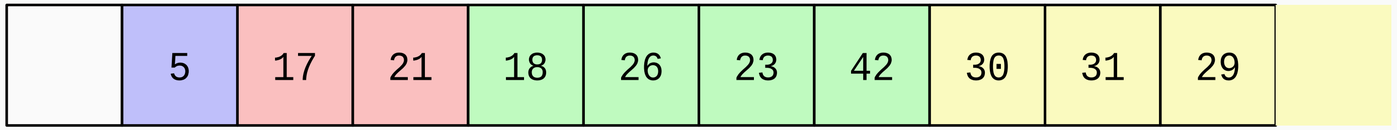


4 elements in array

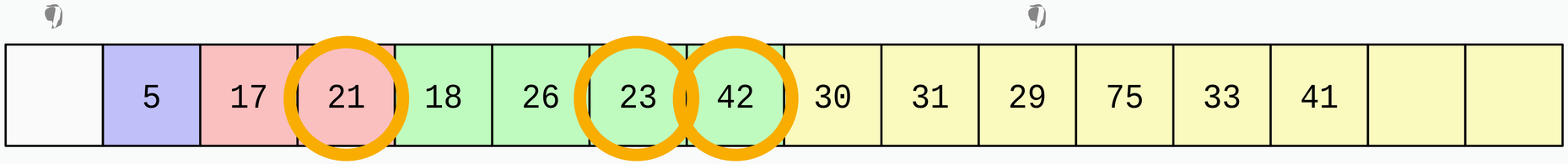
2^2 = 4 nodes in this level

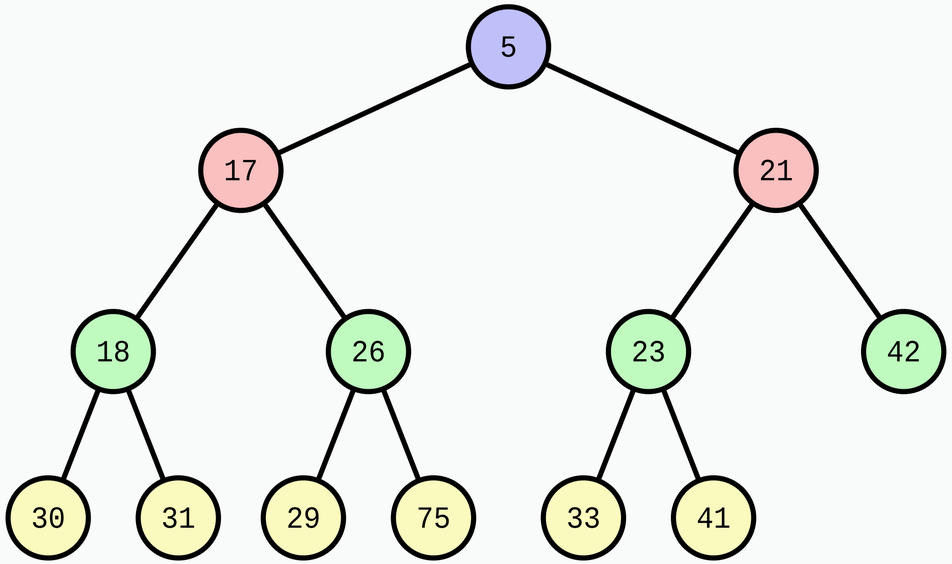
Because a binary heap has a highly regular structure, we can represent it with an

Because a binary heap has a highly regular structure, we can represent it with an .


5
17
21
18
26
23
42
30
31
29

Node at index has children at indices 2 and 2 +1.





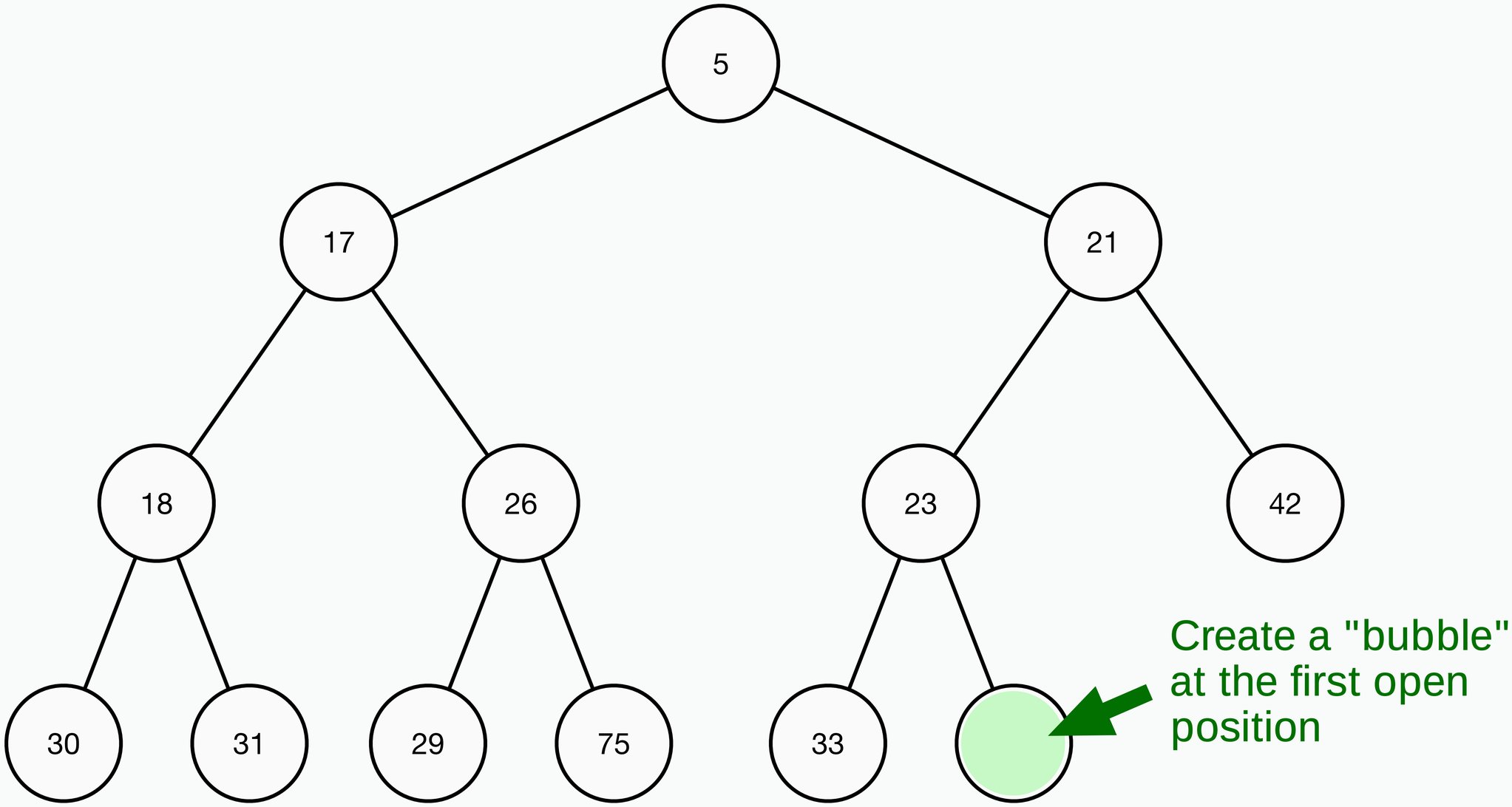
As a static object, a binary heap is of little utility. We need to be able to modify it by inserting and deleting nodes. But we must insert and delete in ways that .

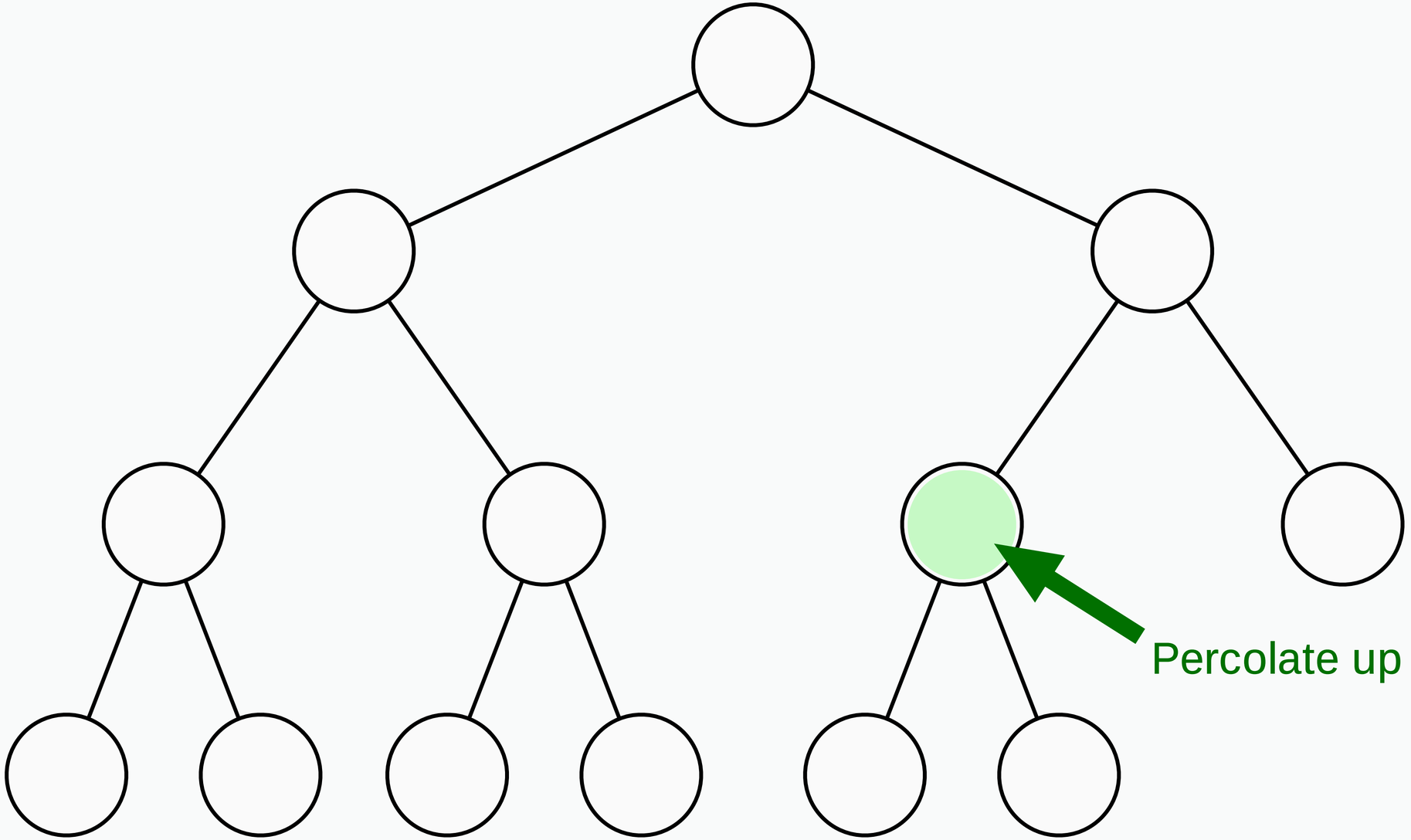
Let's look at the following operations:

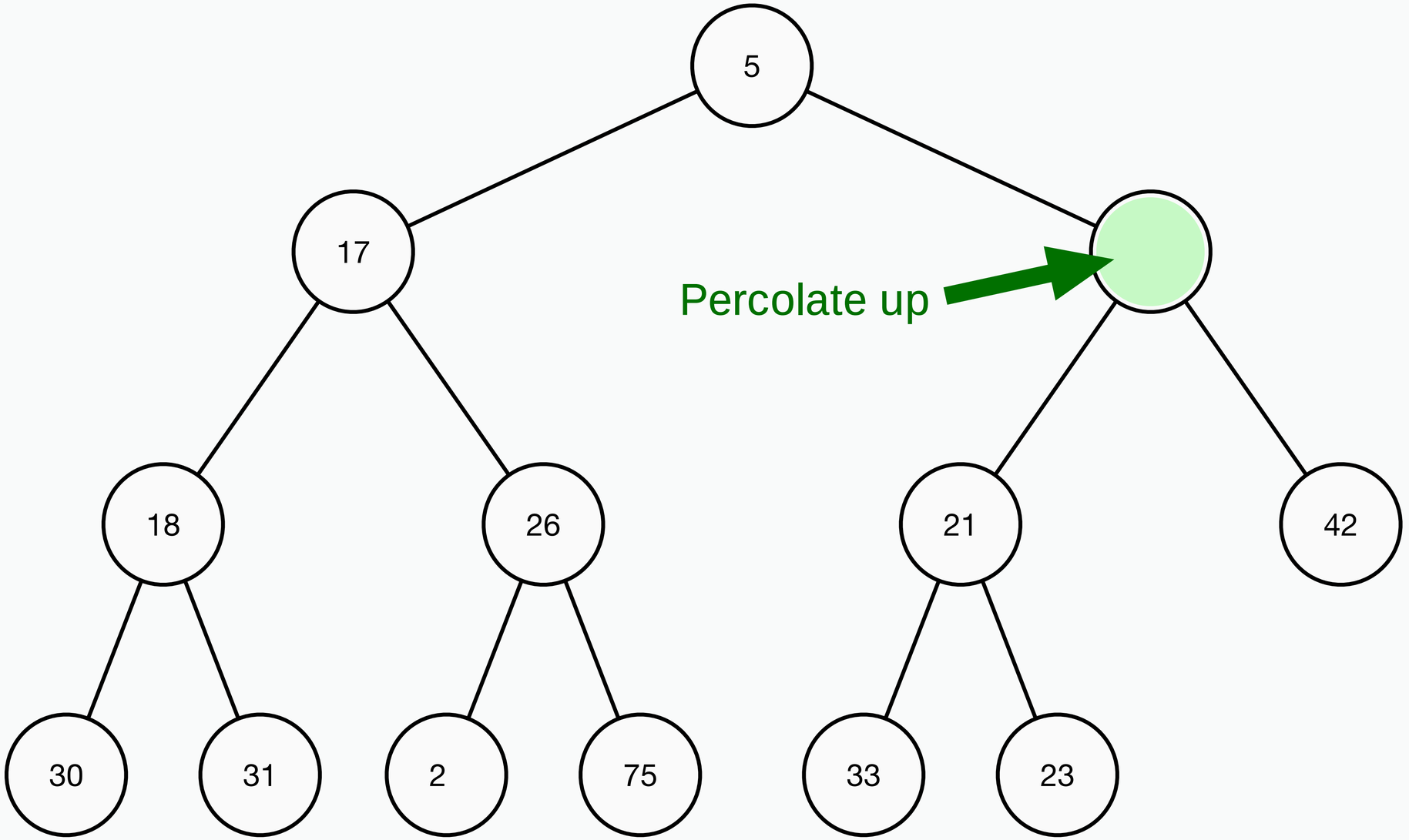
Insert

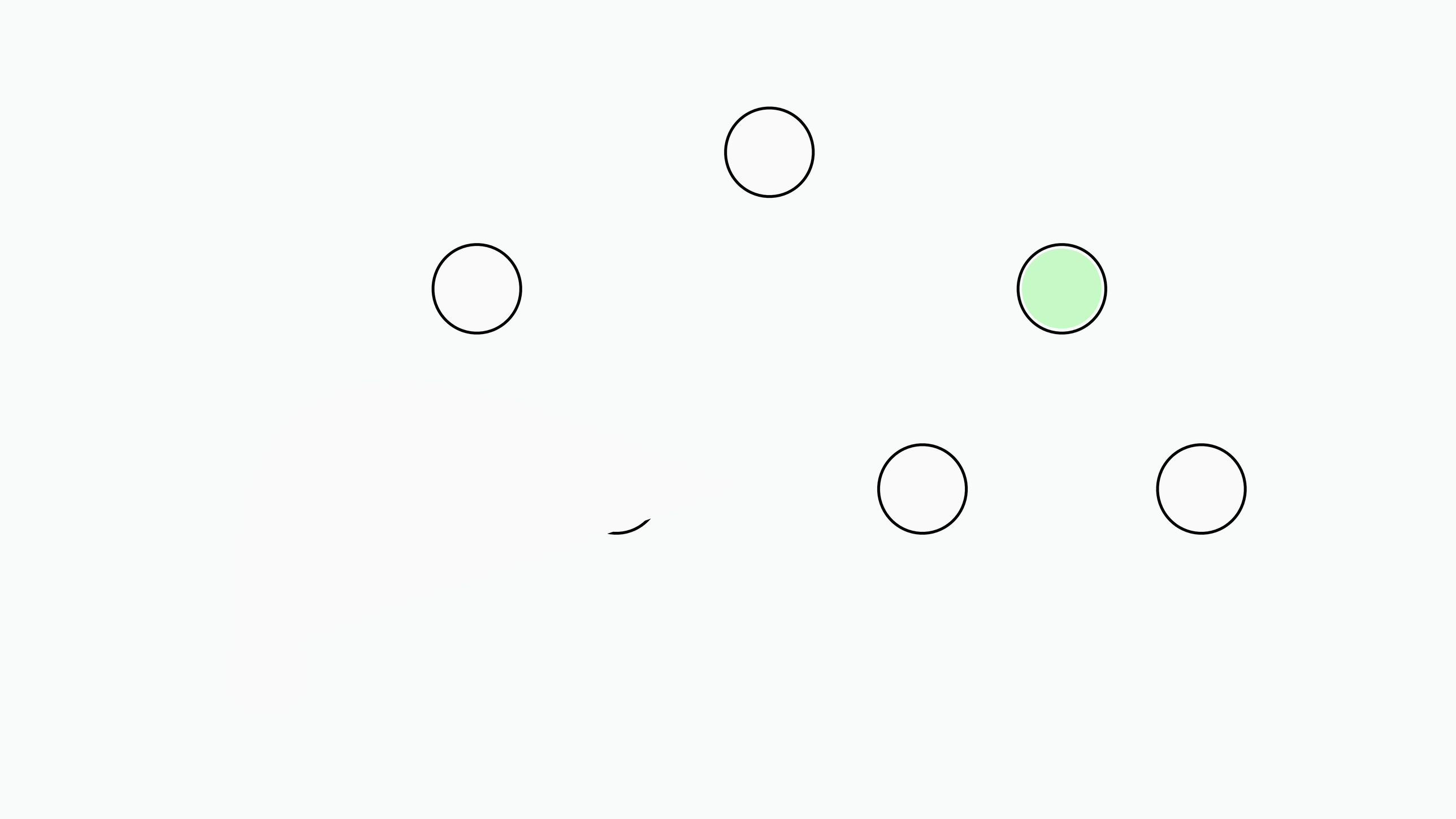
Delete minimum

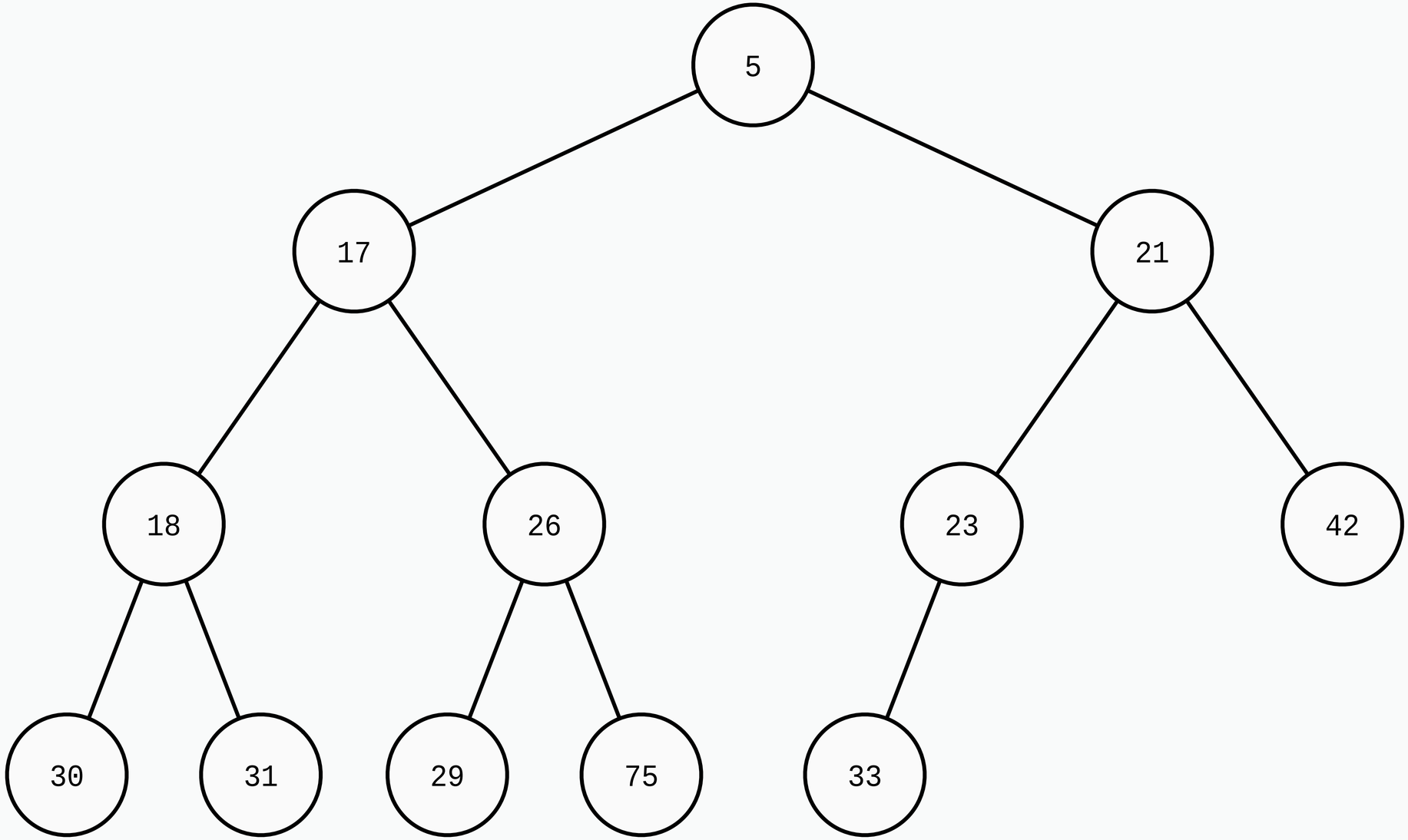
When inserting and deleting we use one of two strategies to preserve structure and heap-order property. We use the metaphor of a "bubble" or within the heap.

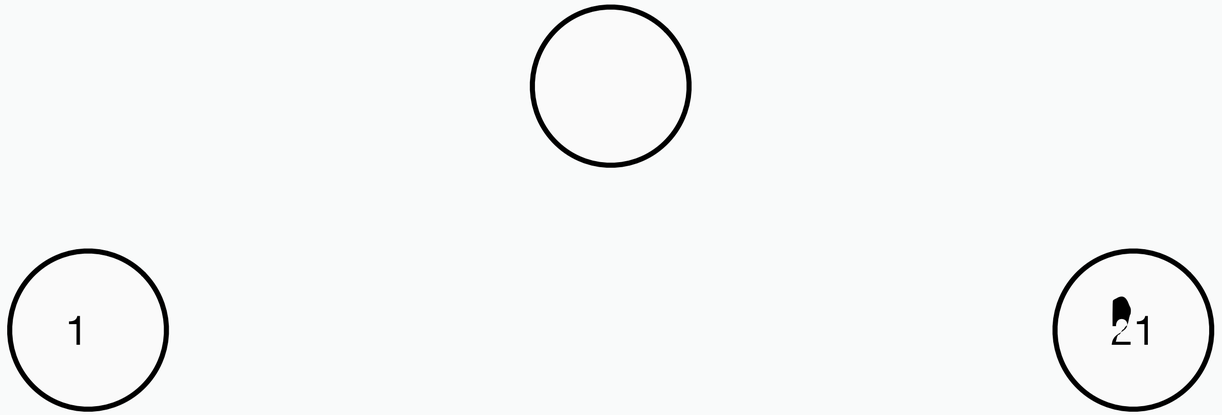

5
17
21
18
26
23
42
30
31
29
75
33
Create a "bubble" at the first open position

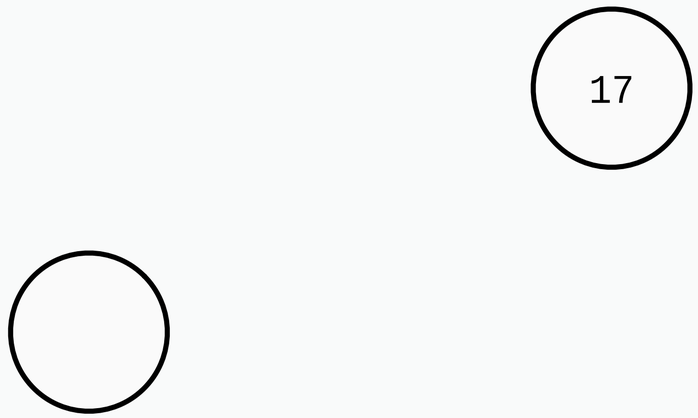

Percolate up

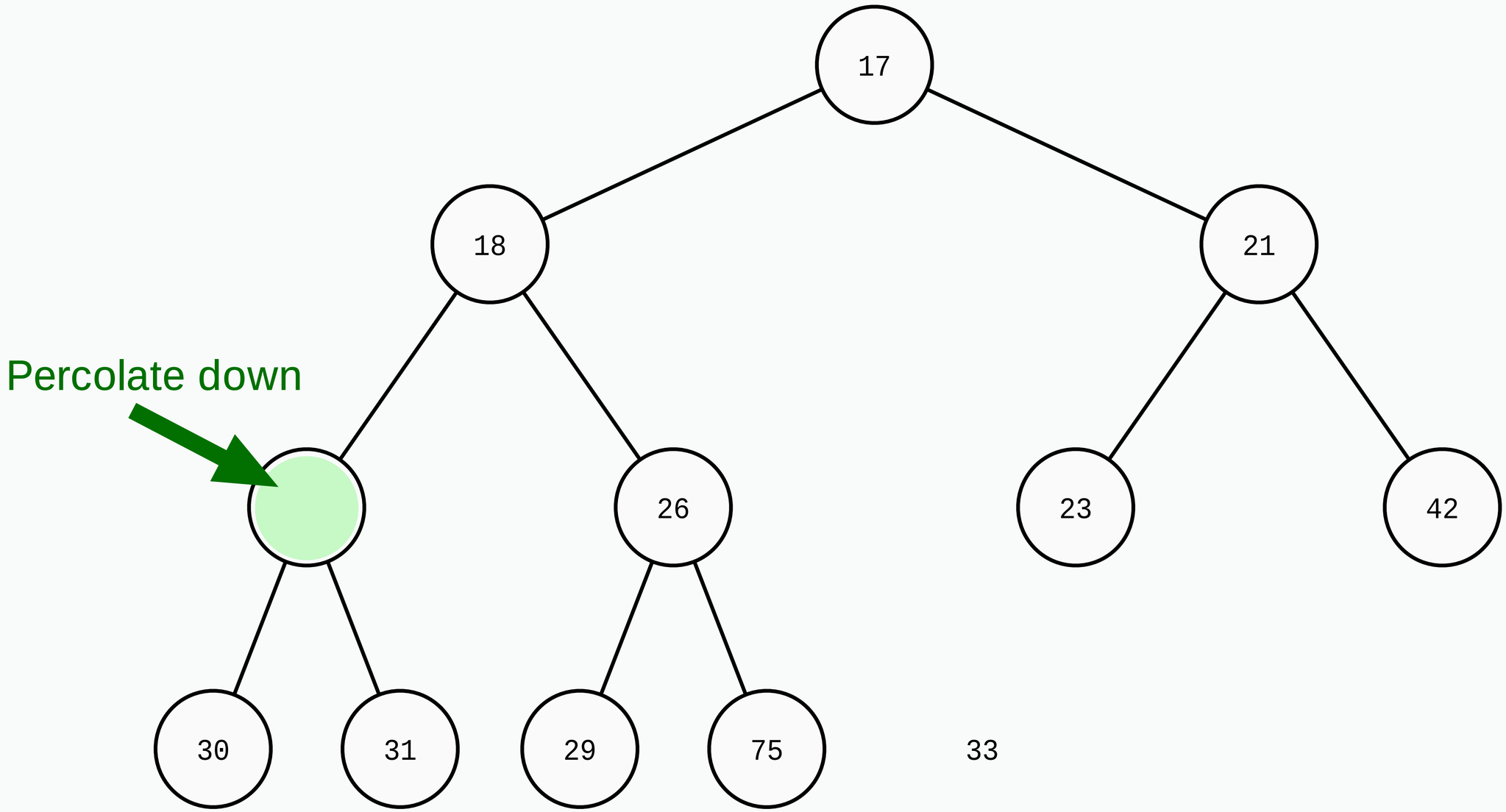

5
17
Percolate up
18
26
21
42
30
31
2
75
33
23

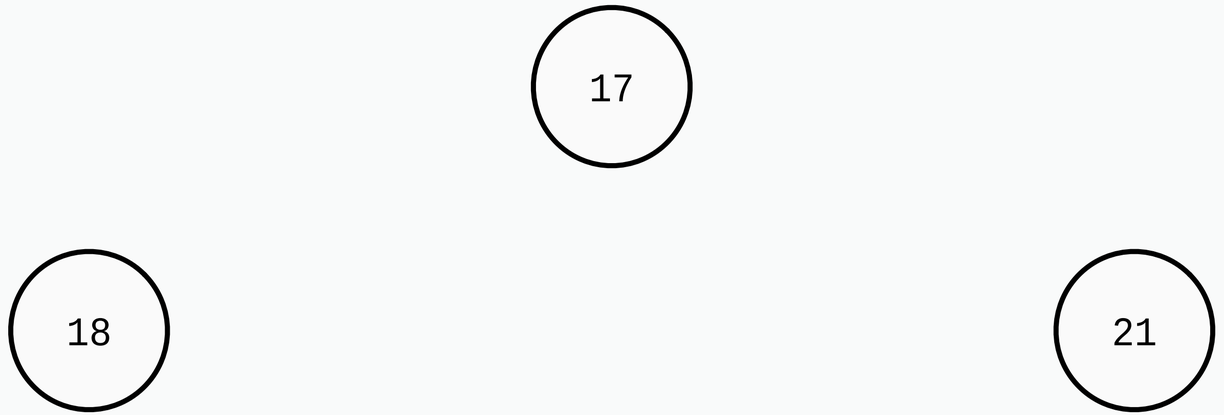


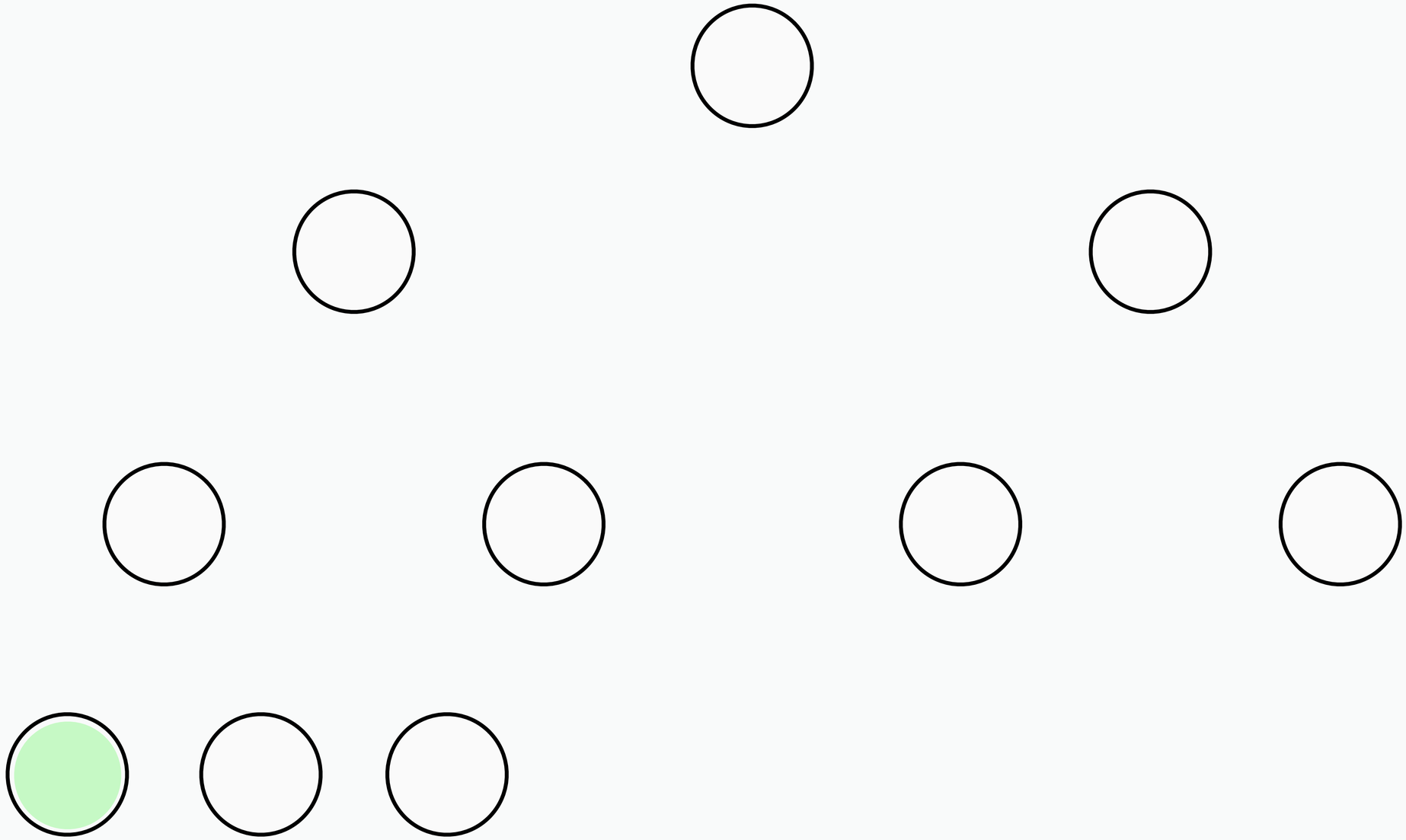

5
17
21
18
26
23
42
30
31
29
75
33

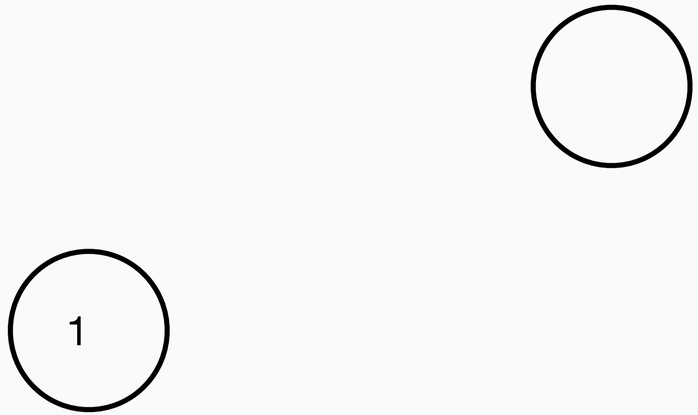

1
21

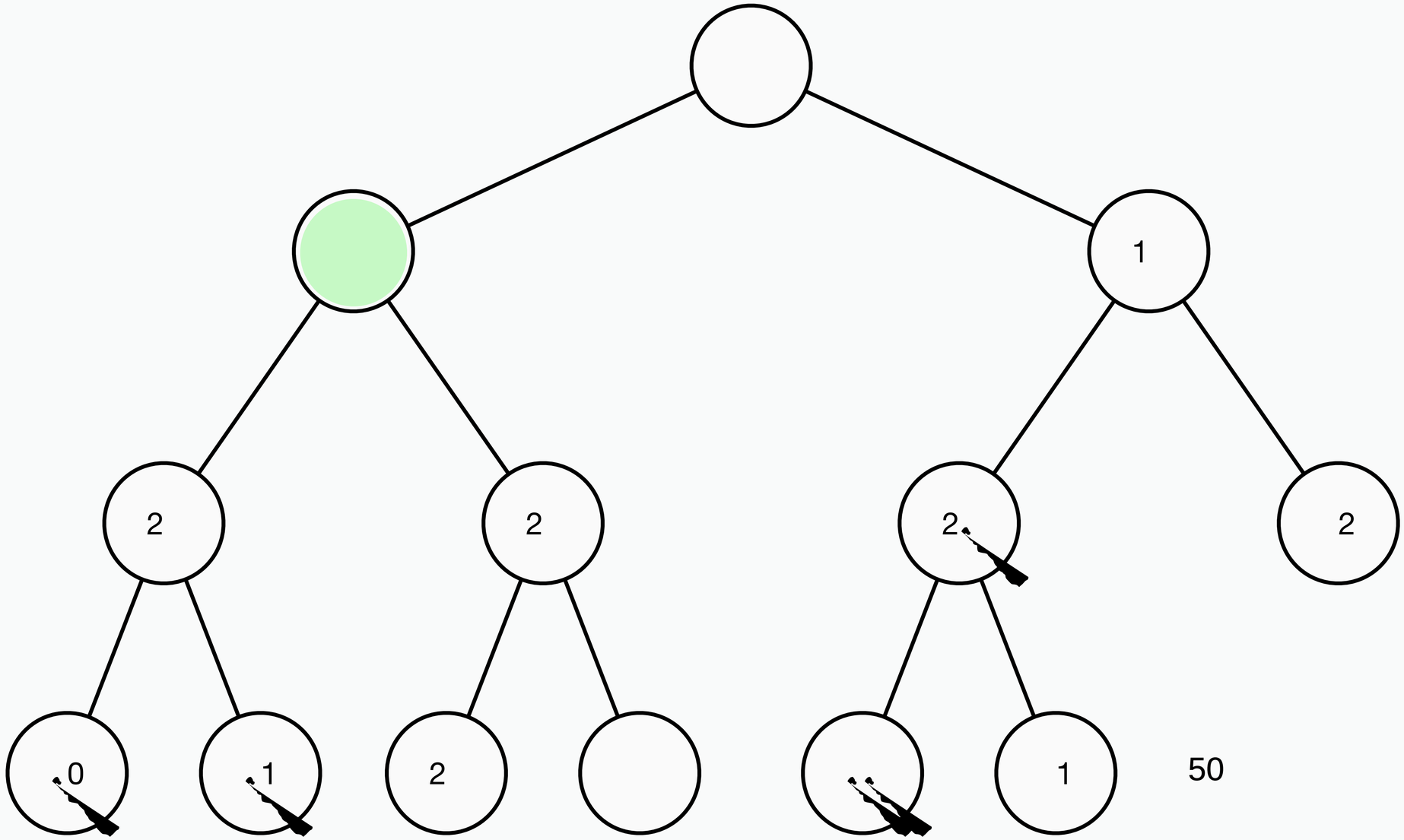

17


17
18
21
Percolate down
26
23
42
30
31
29
75
33


17
18
21




1


1
2
2
2
2
0
1
2
1
50

Binary heap is a binary tree with the -.

Structure property and heap-order property must be preserved.

Binary heap can be represented with an .

We modify the heap by creating and
until a new (or orphaned) value can find a suitable node in the tree.

Insert and delete operations have $\mathcal{O}($) complexity.

Binary heaps are well-suited to and other applications.