B-Trees 

set the stage

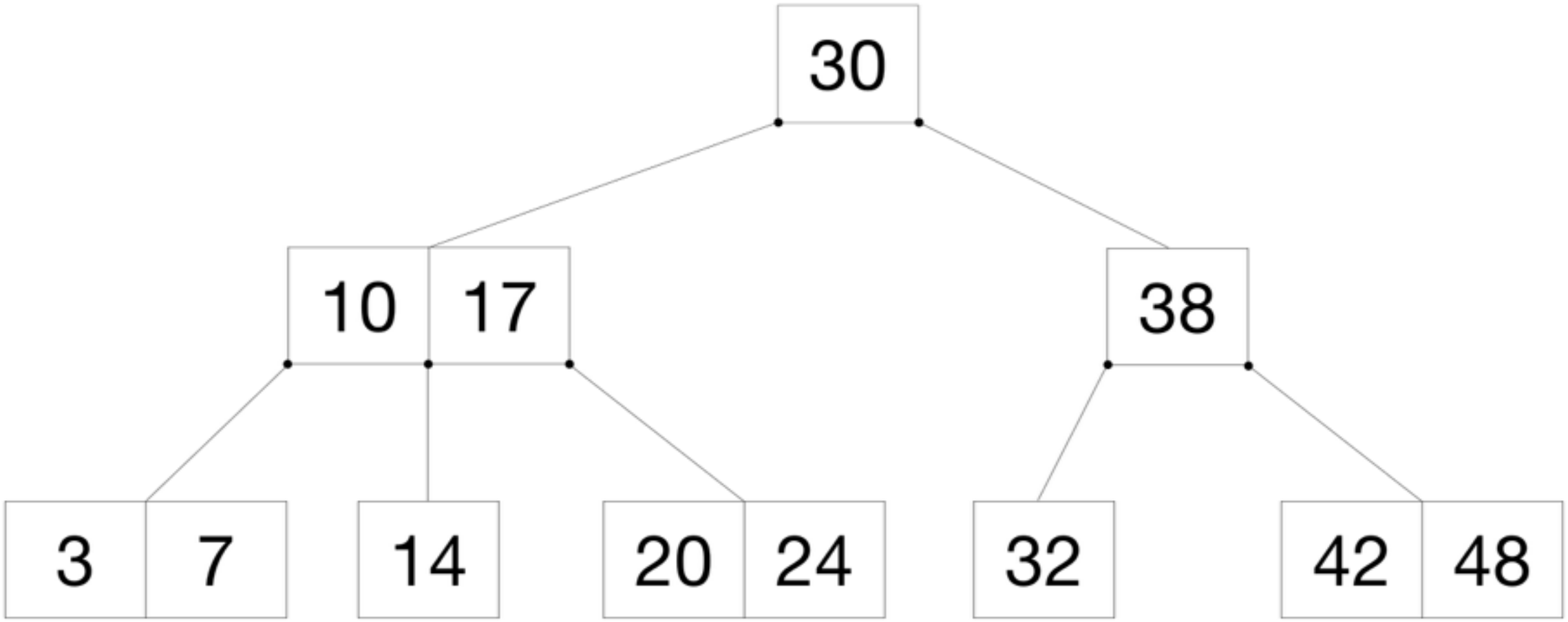
memory access time

RAM vs. virtual memory (on disk)

why does this even matter?

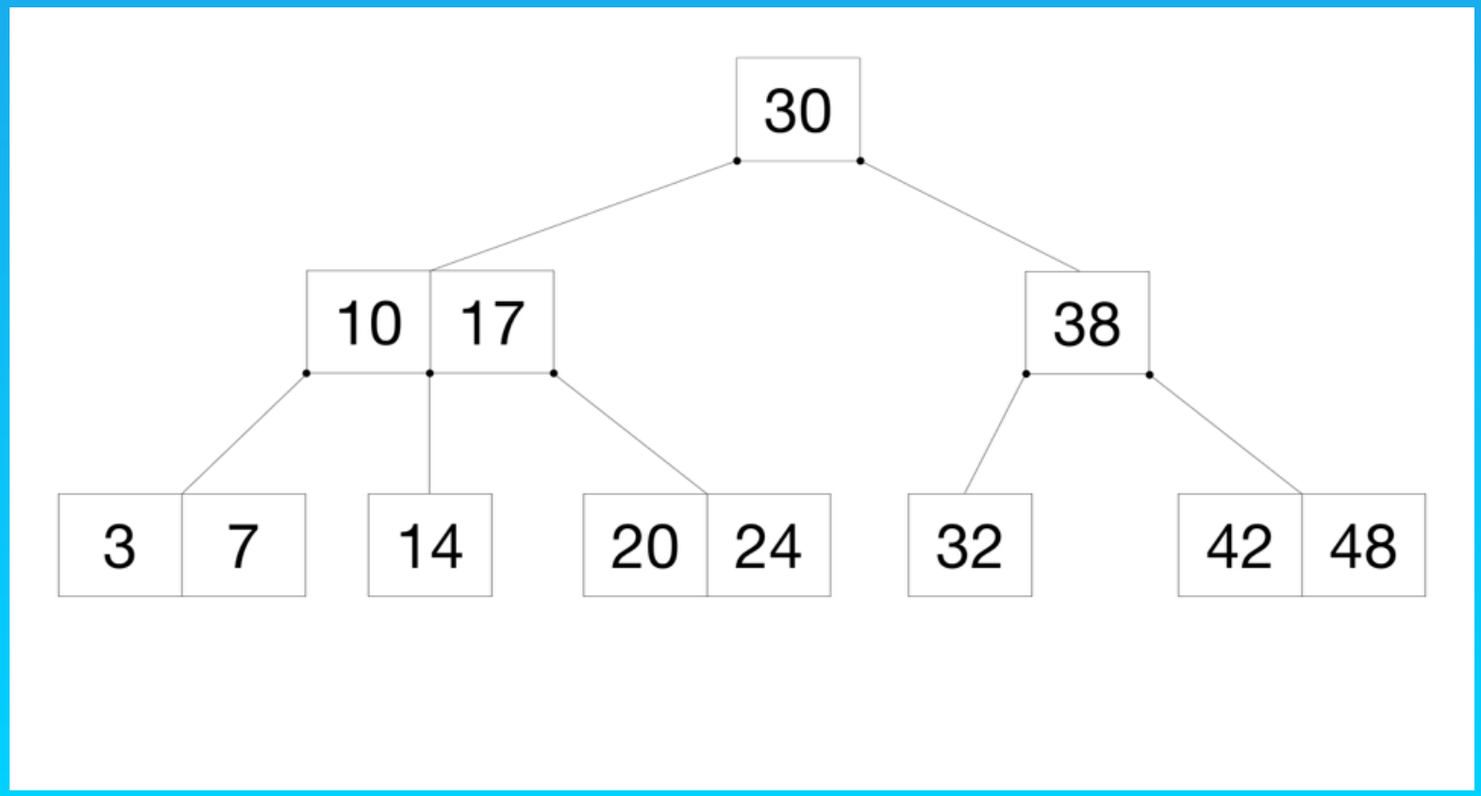
most of the tree operations (search, inset, delete, max, min, etc.) require O(h) disk accesses where h is height of the tree.

example



characteristics

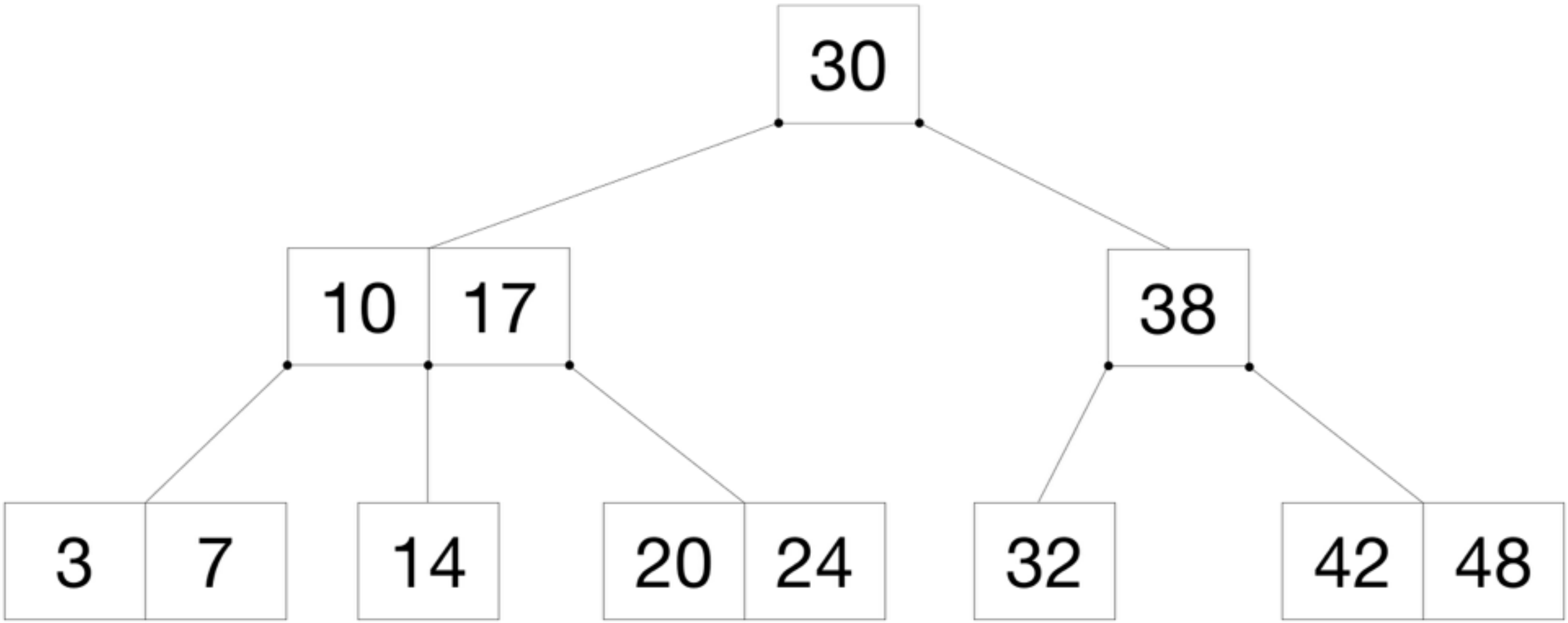
1. All leaves are at same level
2. A B-Tree is defined by the term *minimum degree* B. The value of b depends upon disk block size.
3. All nodes except root must contain at least b-1 keys. Root may contain minimum 1 key.
4. All nodes (including root) must contain less than 2b - 1 keys.
5. Number of children of a node is equal to the number of keys in it plus 1.
6. All keys of a node are sorted in increasing order. The child between two keys k1 and k2 contains all keys in range from k1 and k2.
7. Like other balanced BSTs, time complexity to search, insert and delete is O(log(n))



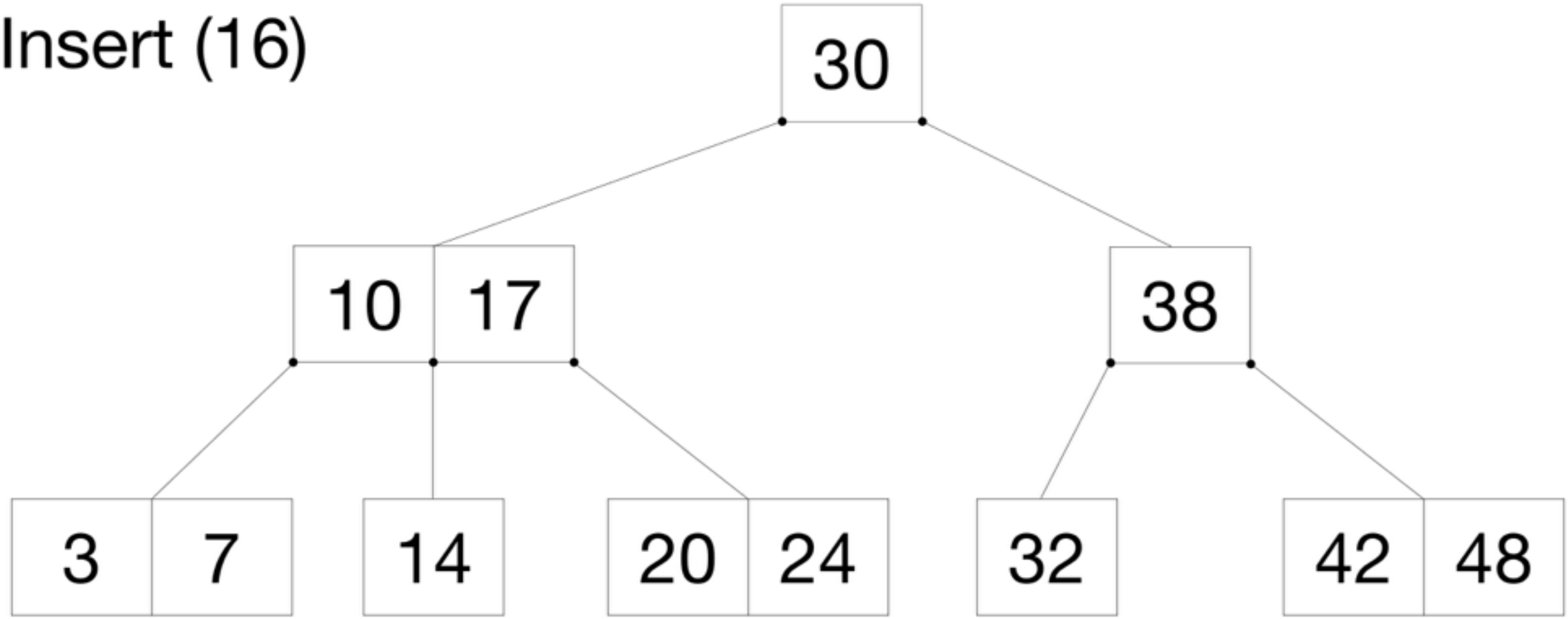
“From a practical point of view, B-trees, therefore, guarantee an access time of less than 10 ms even for extremely large datasets.”

— Dr. Rudolf Bayer, co-inventor of the B-tree

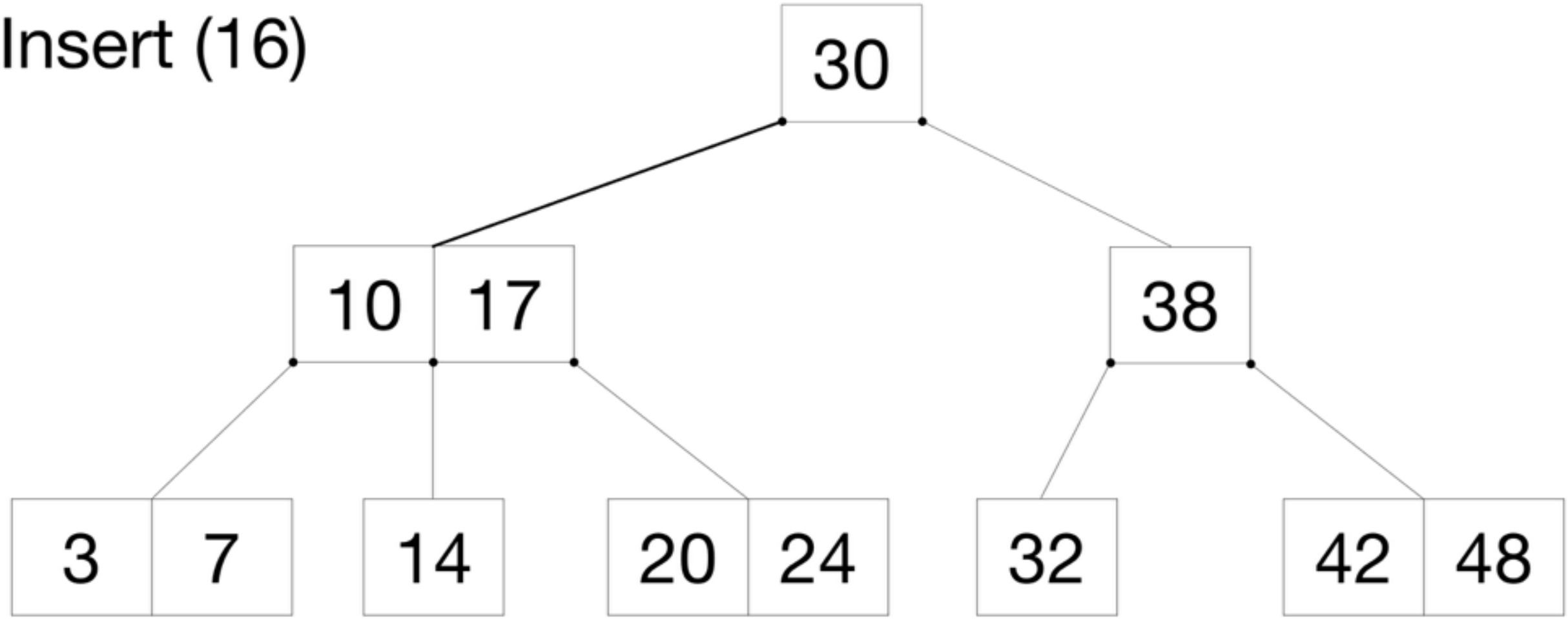
insert


30
10
17
38
3
7
14
20
24
32
42
48

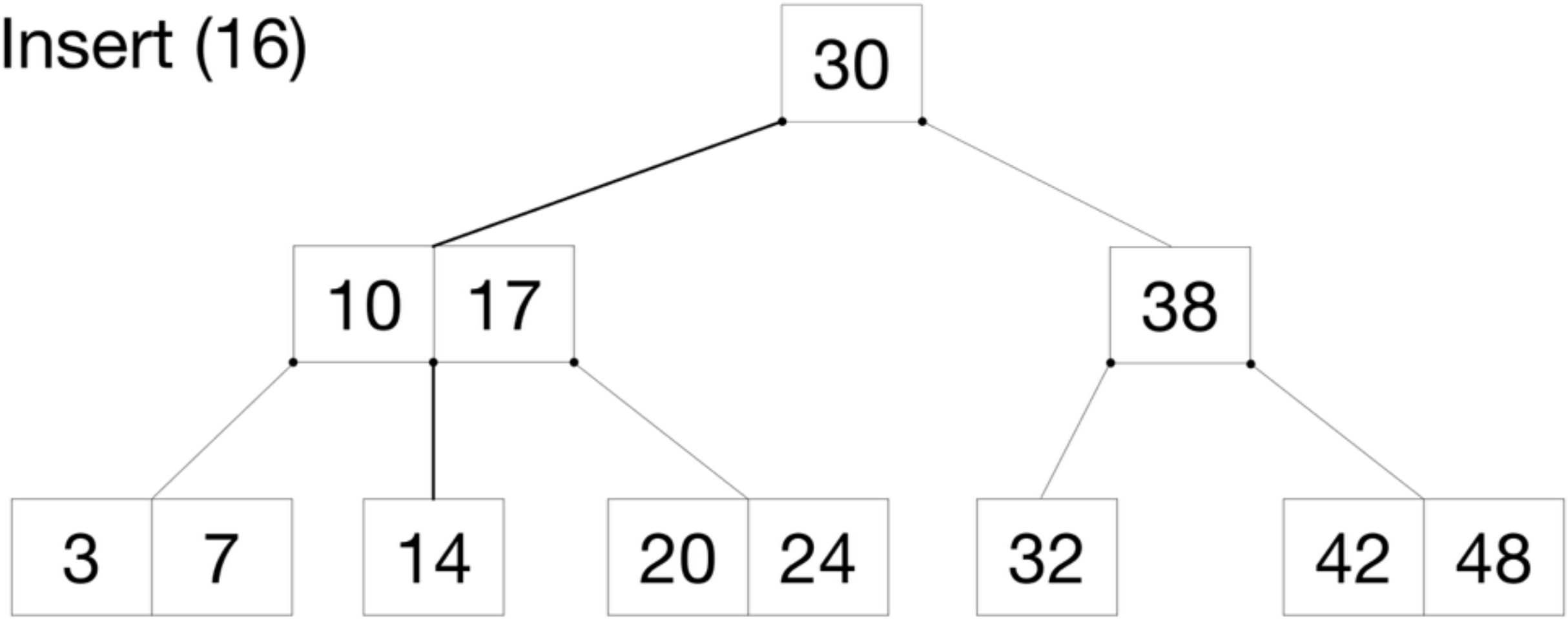
Insert (16)



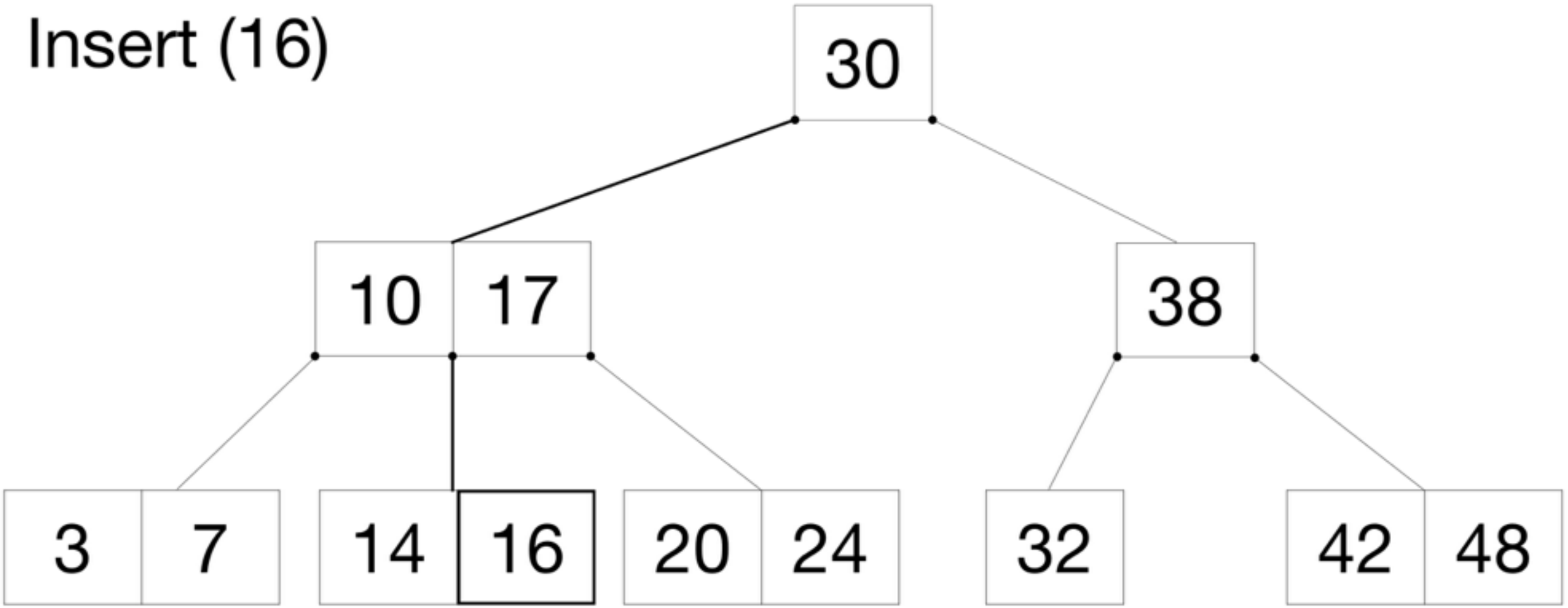
Insert (16)



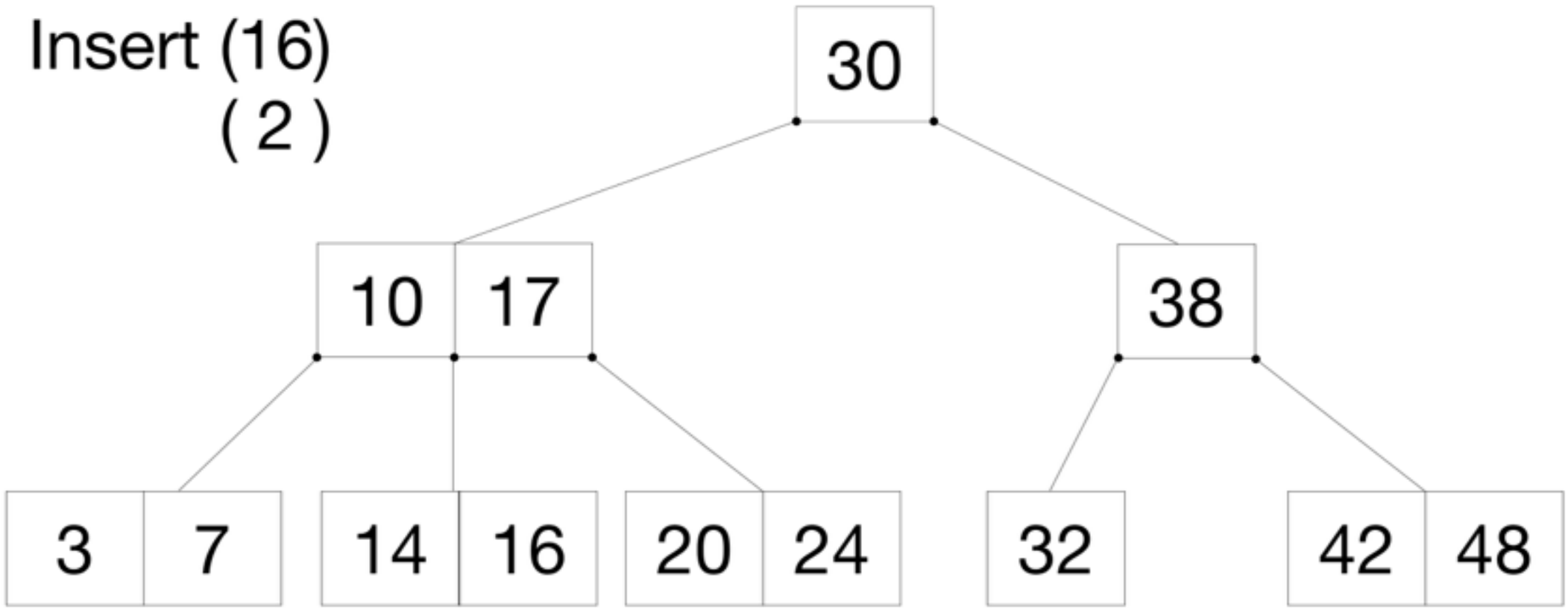
Insert (16)



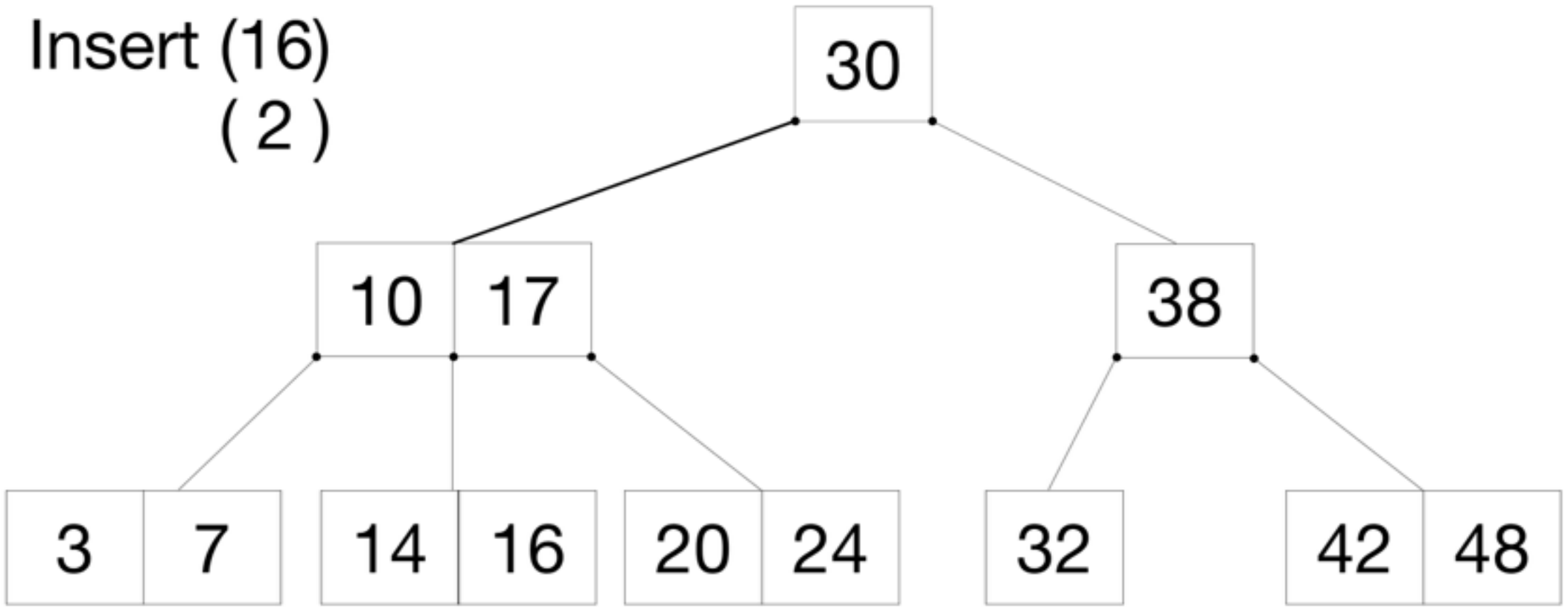
Insert (16)



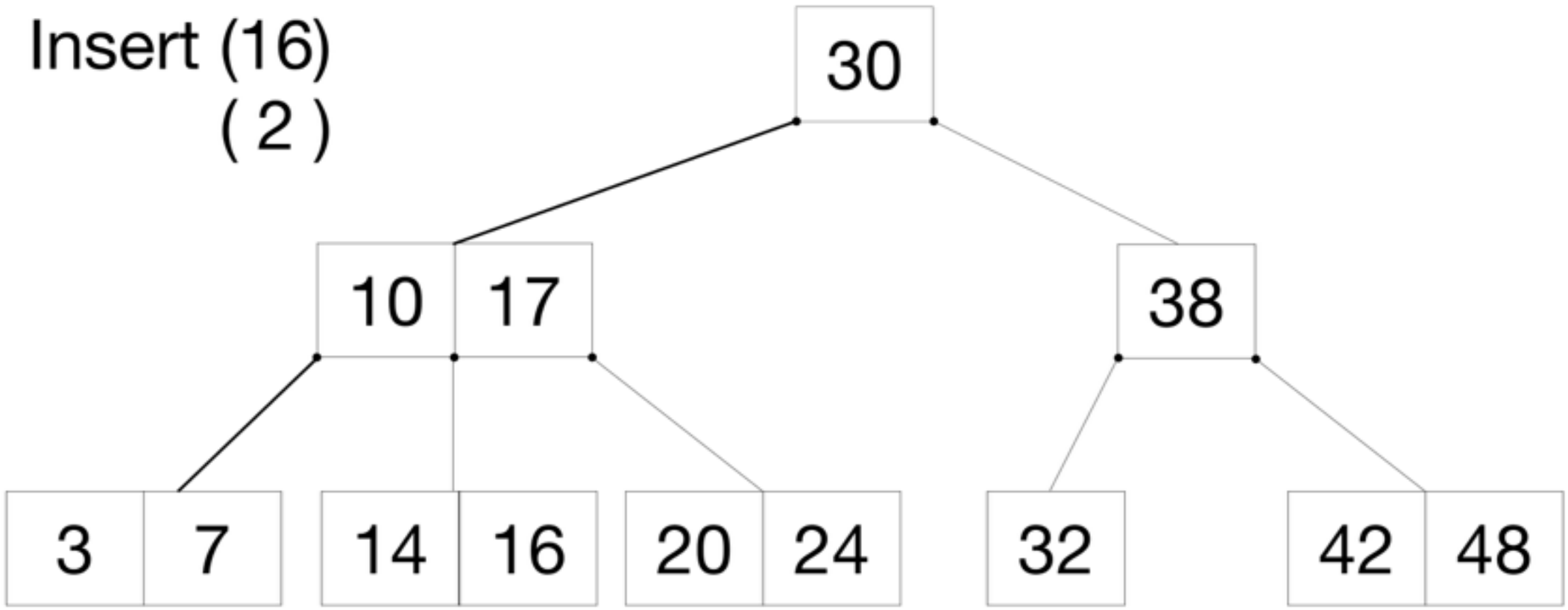
Insert (16)
(2)



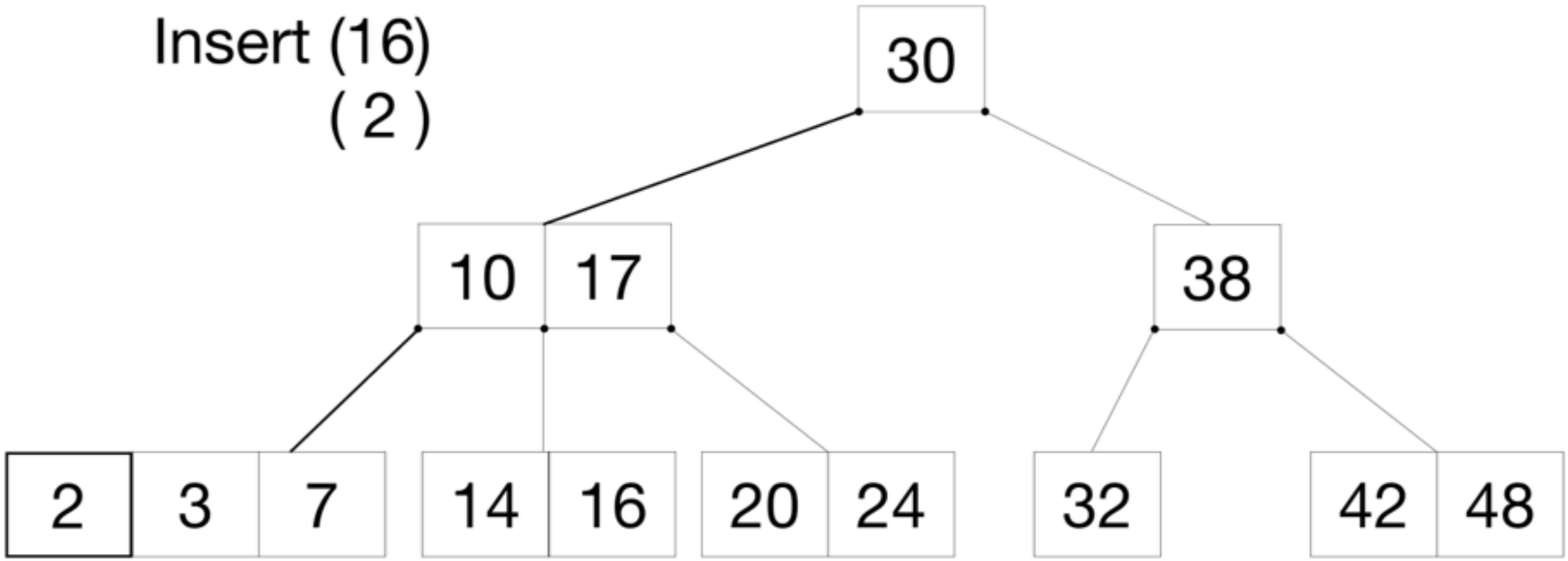
Insert (16)
(2)



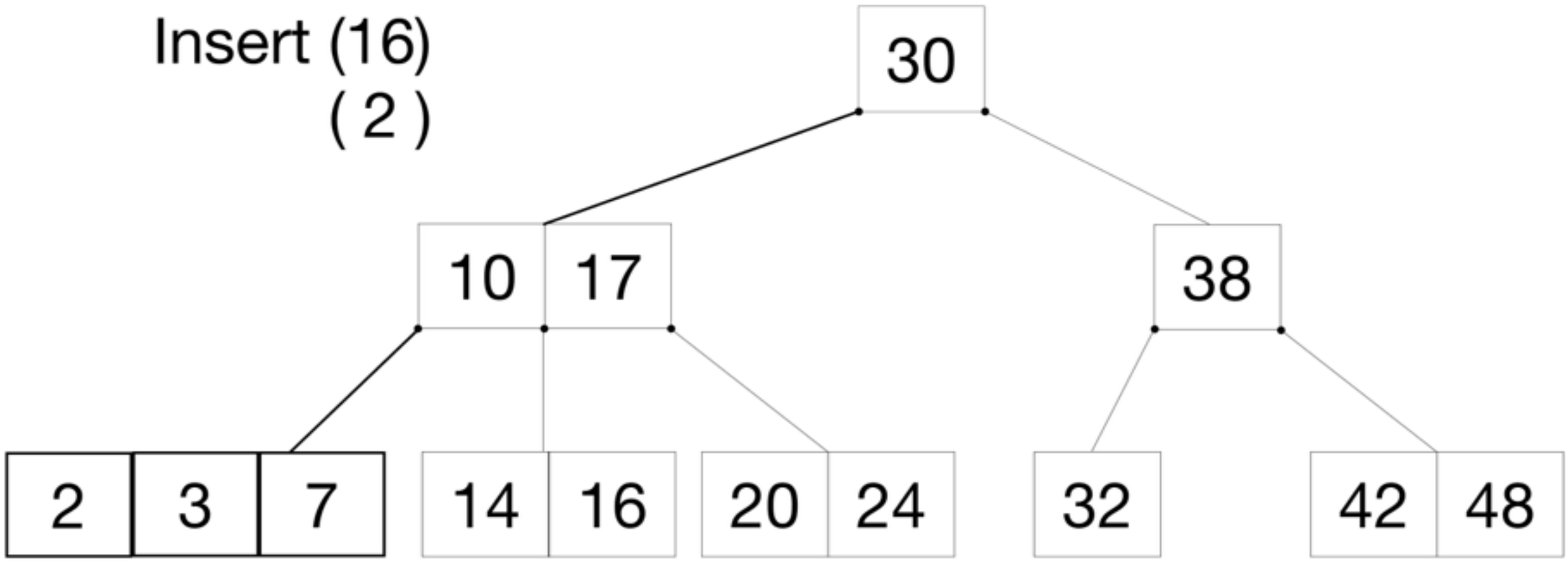
Insert (16)
(2)



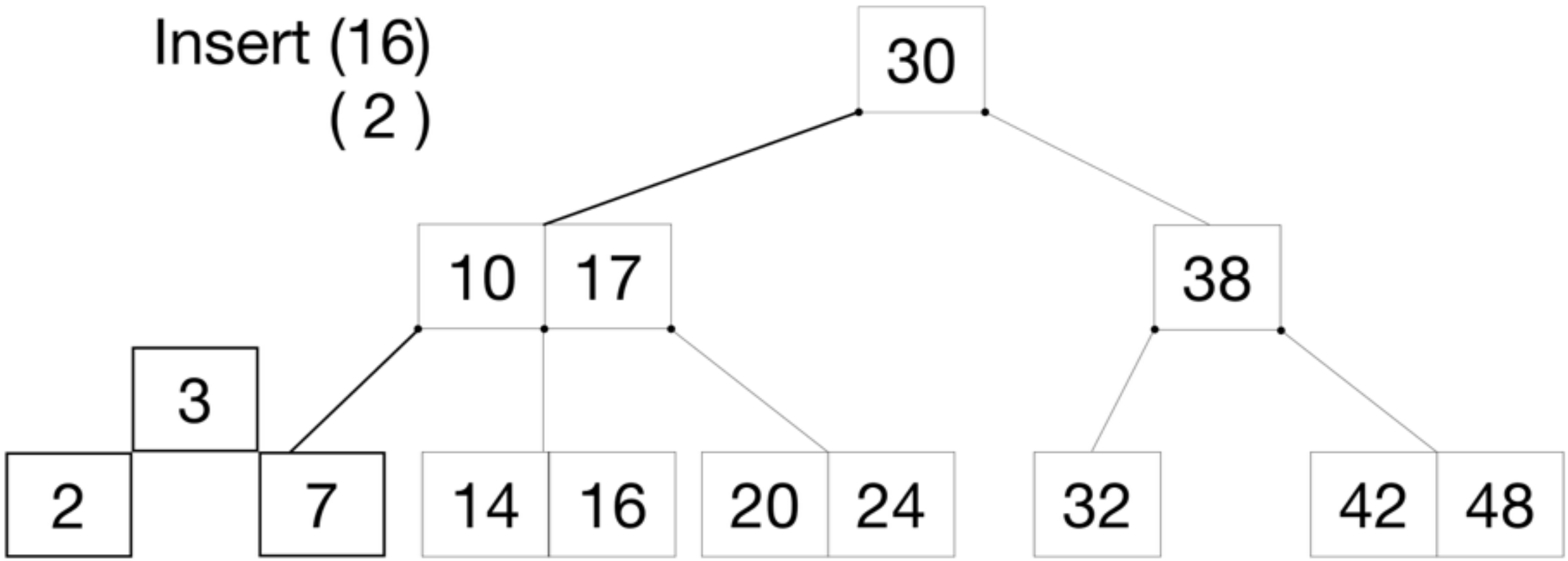
Insert (16)
(2)



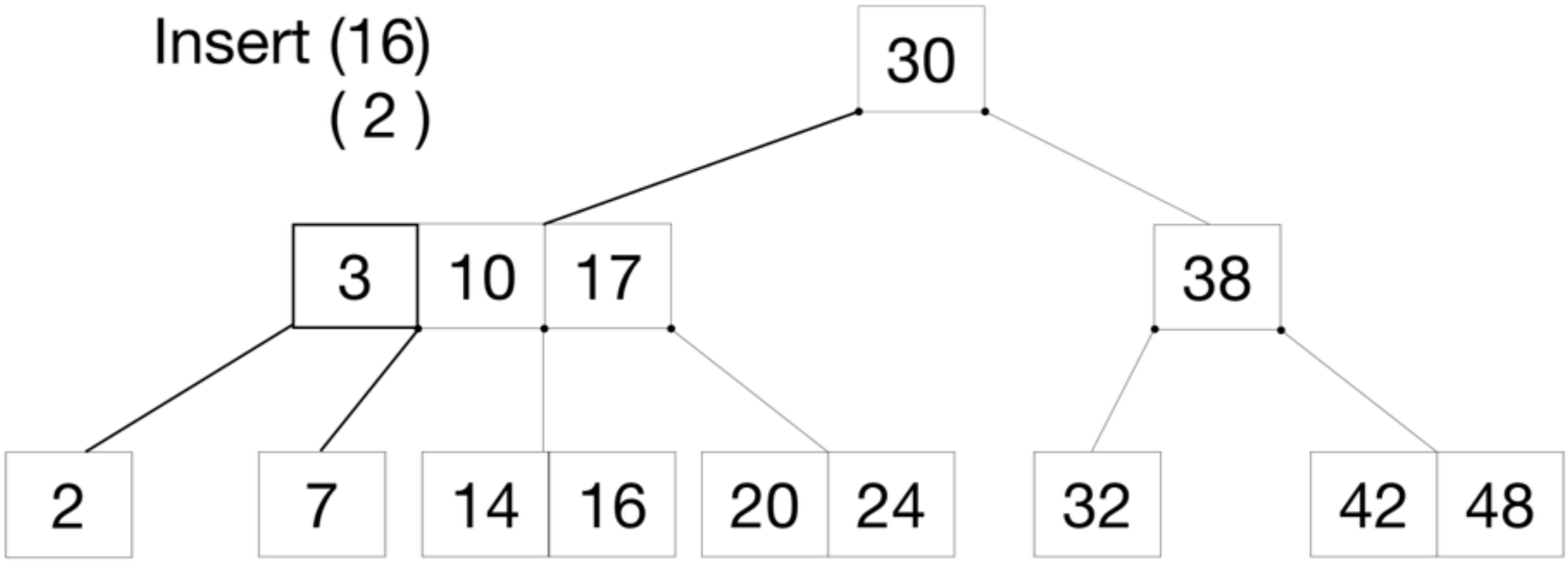
Insert (16)
(2)



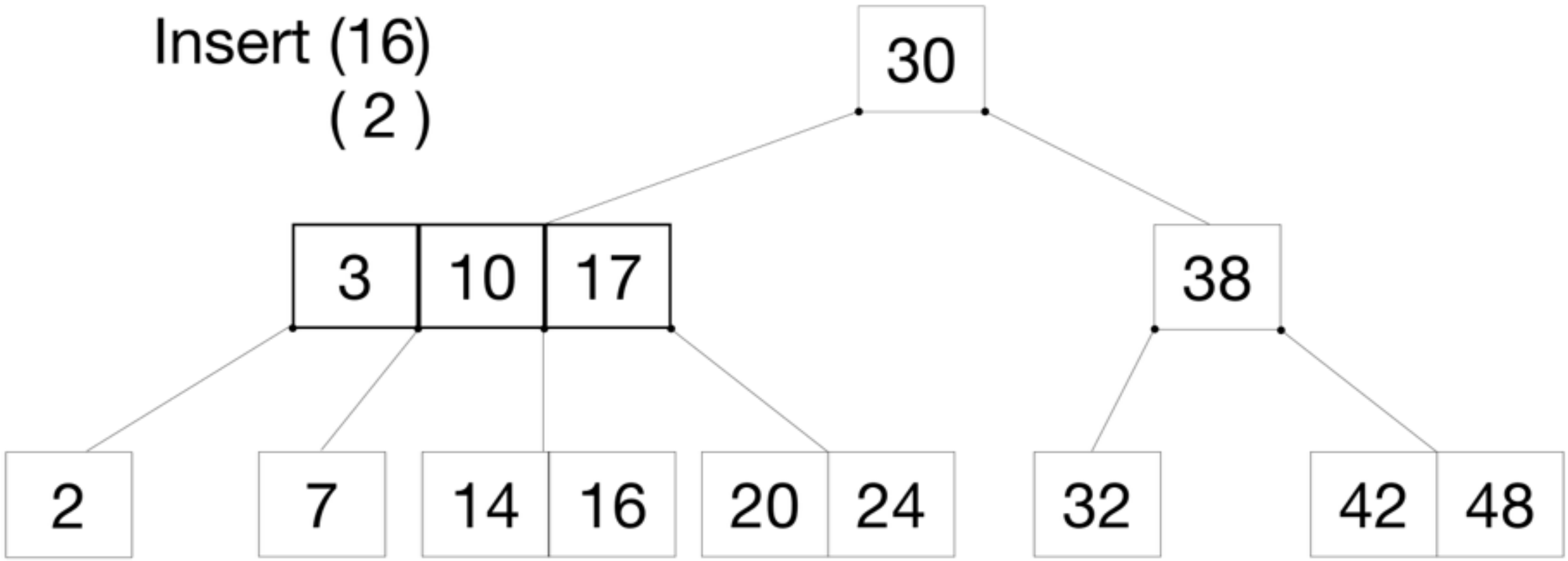
Insert (16)
(2)



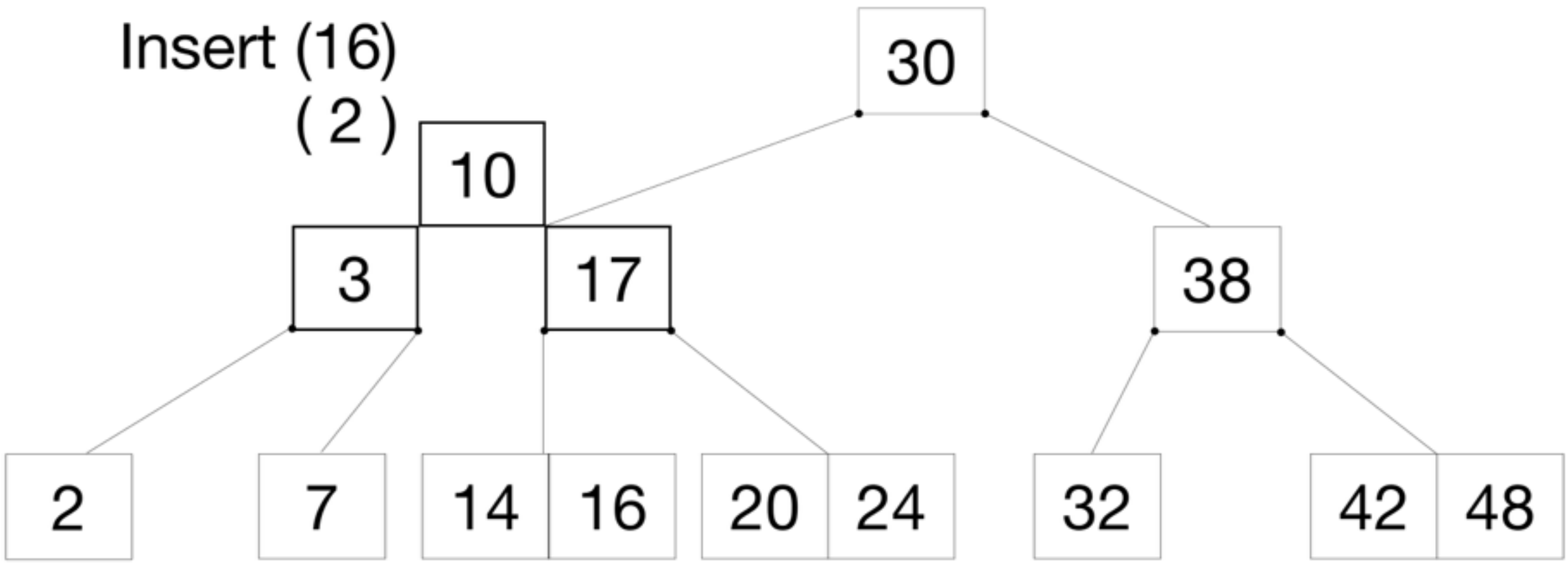
Insert (16)
(2)

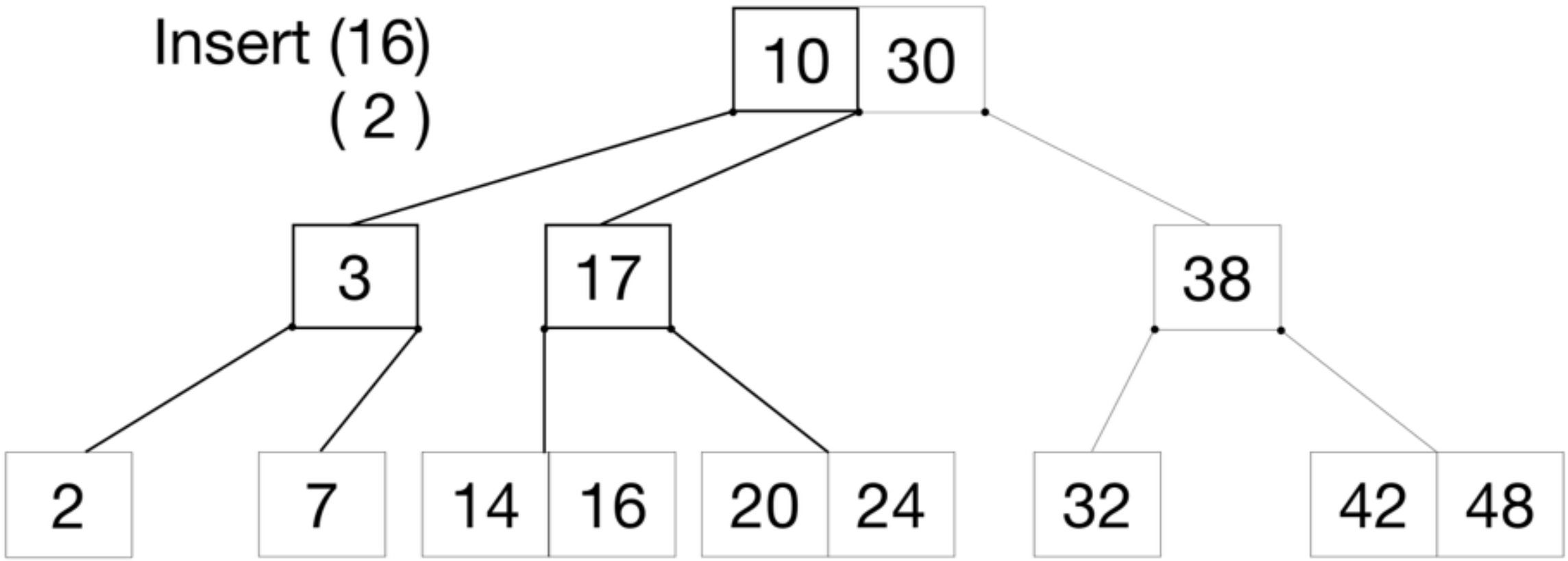


Insert (16)
(2)

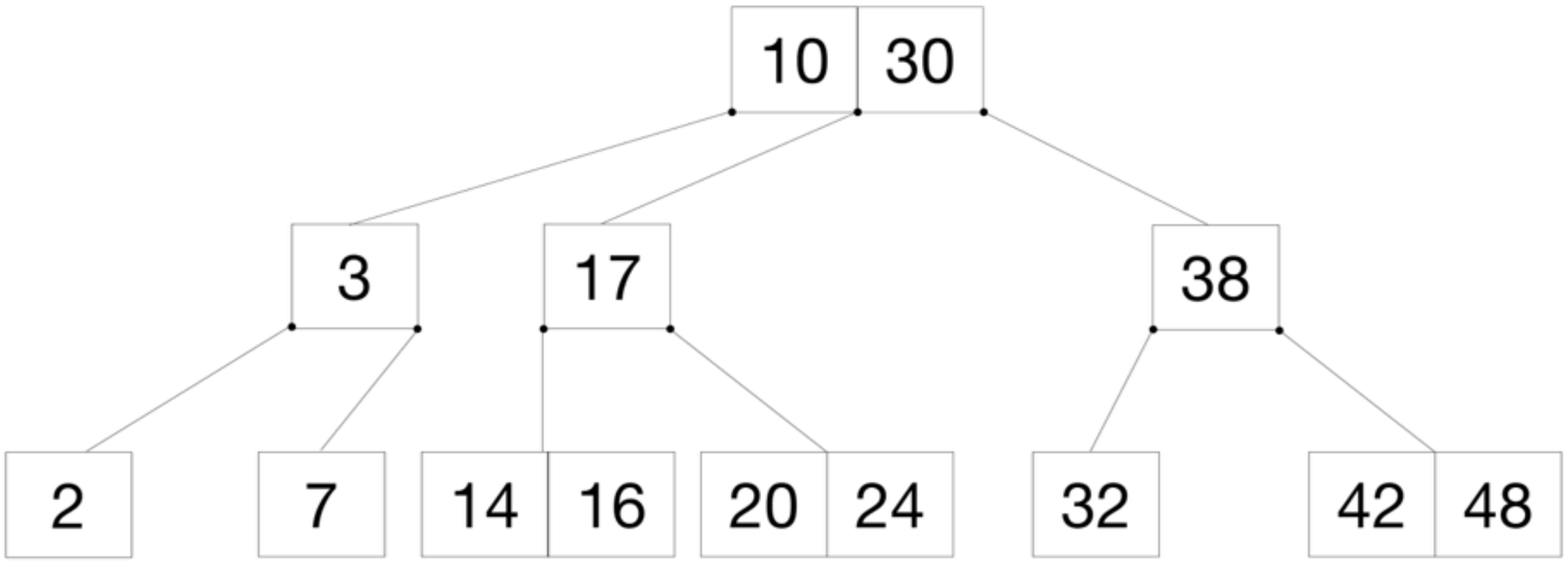


Insert (16)

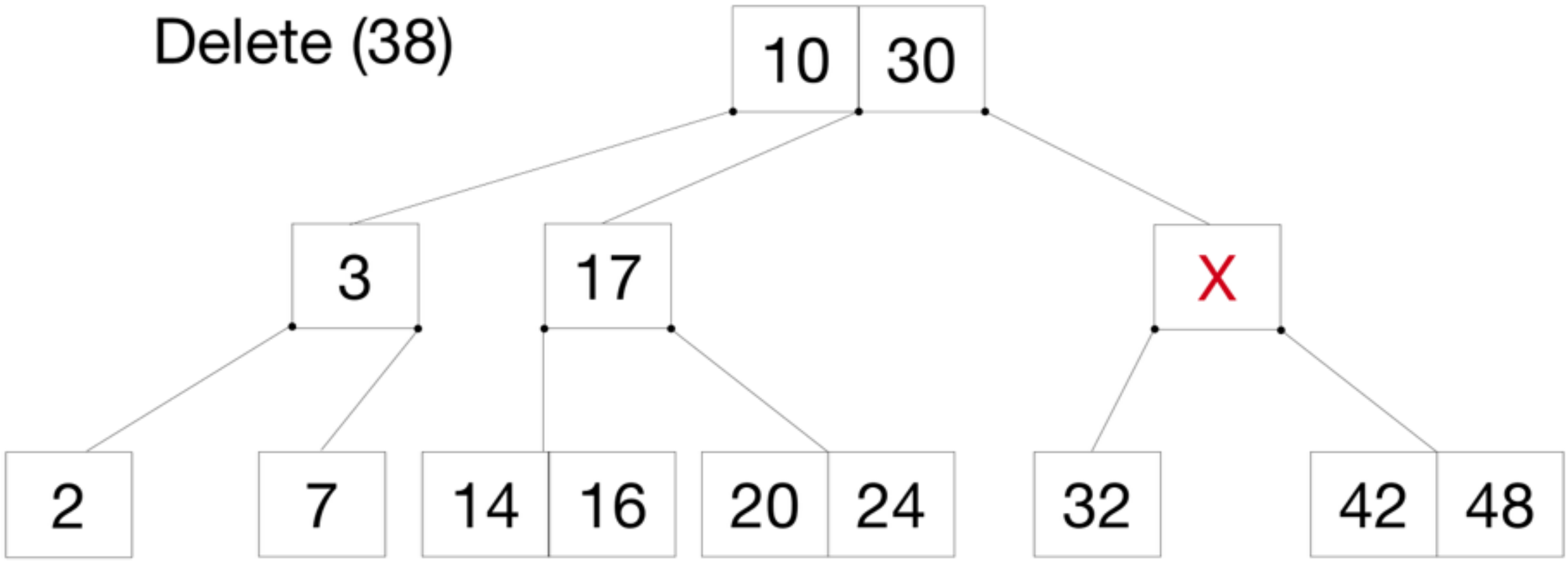



Insert (16)
(2)
10
30
3
17
38
2
7
14
16
20
24
32
42
48

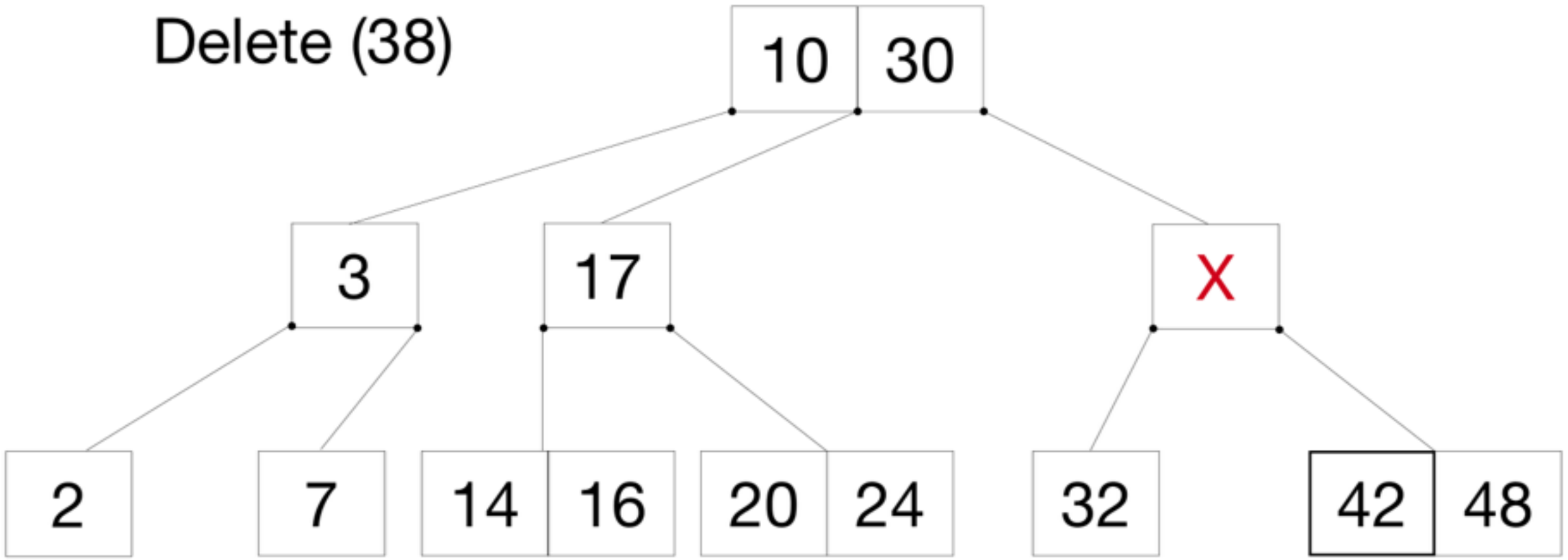
delete


10
30
3
17
38
2
7
14
16
20
24
32
42
48

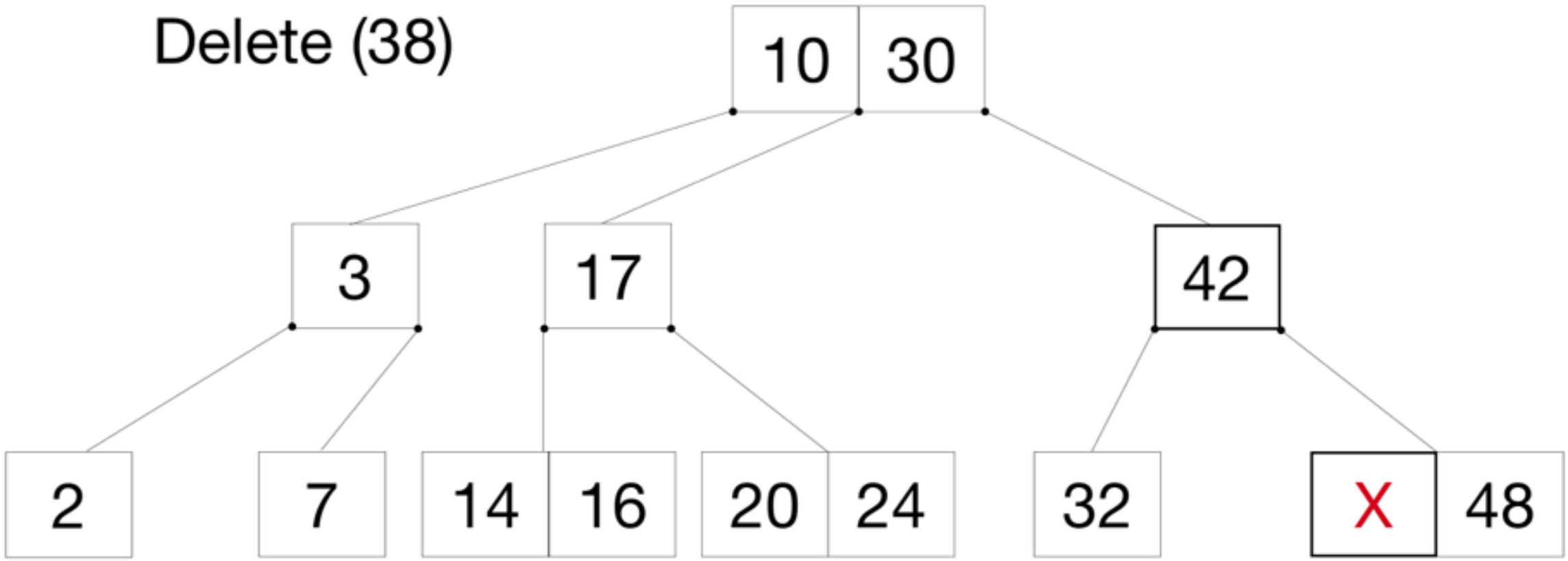
Delete (38)



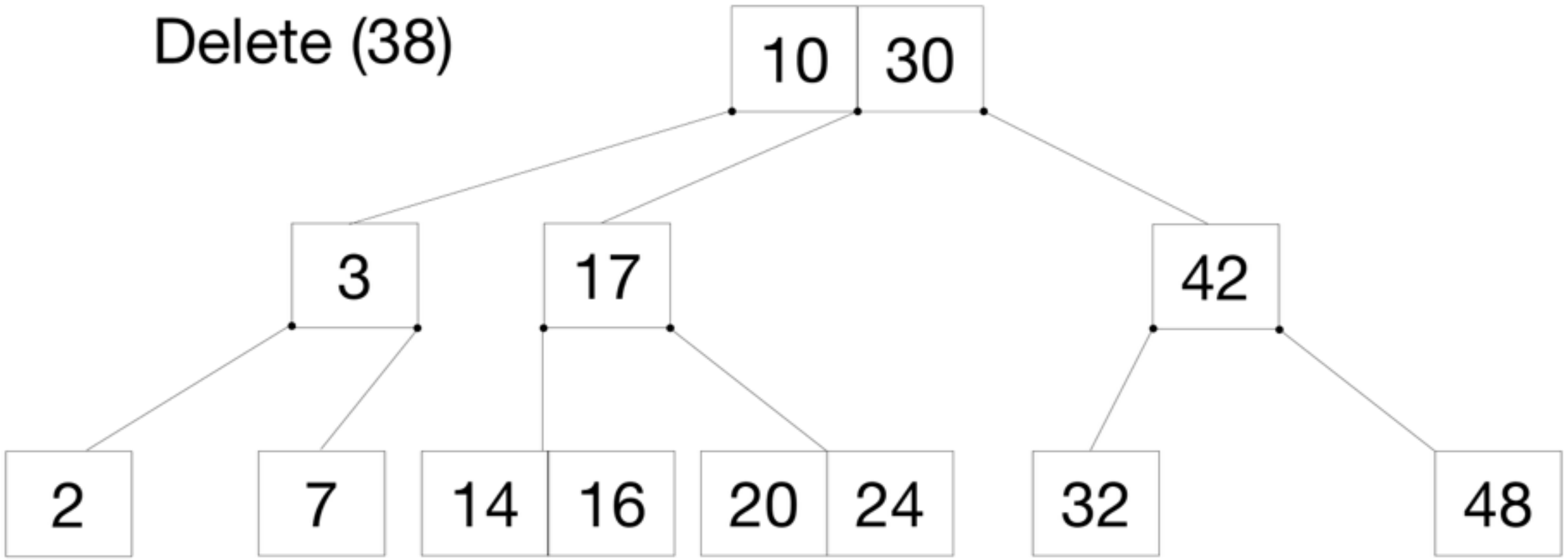
Delete (38)



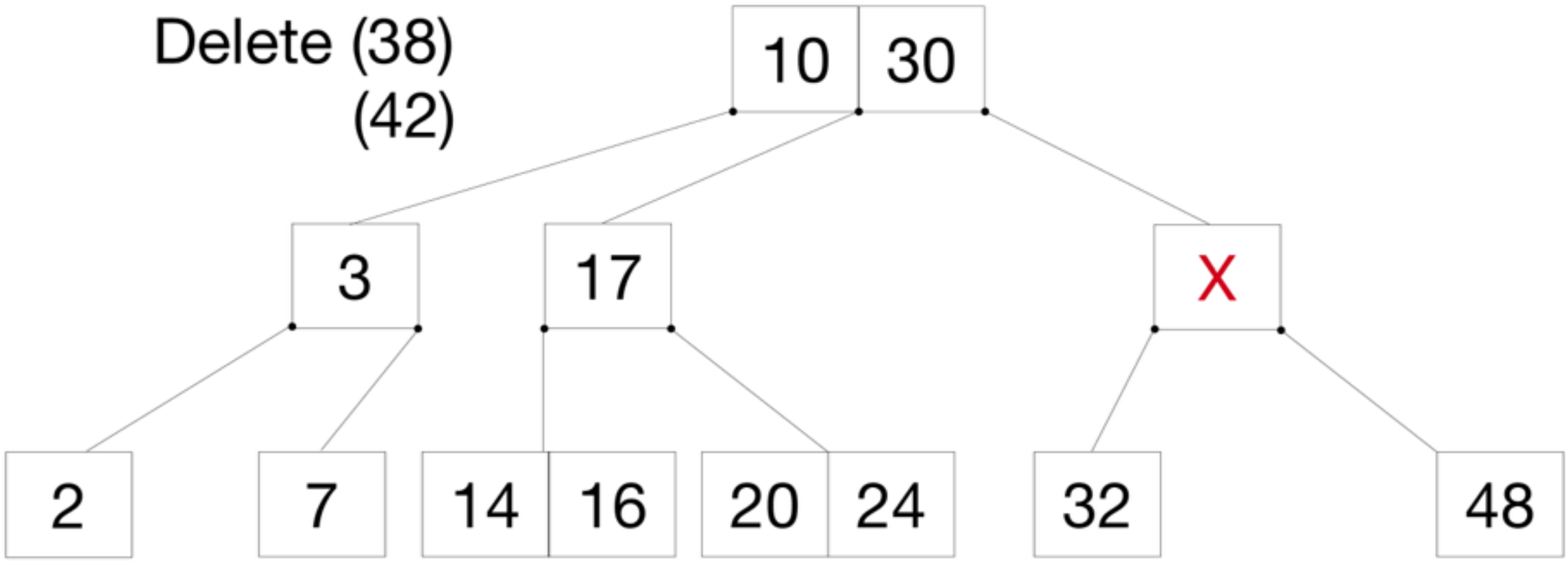
Delete (38)



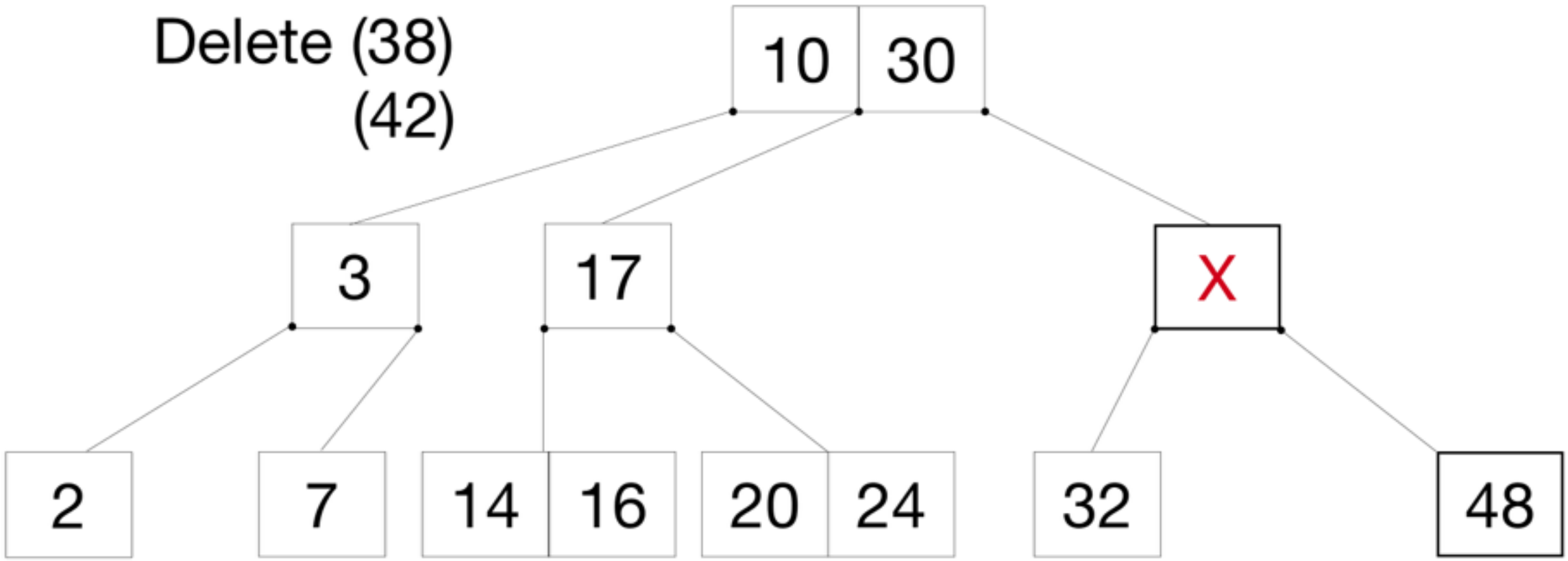
Delete (38)



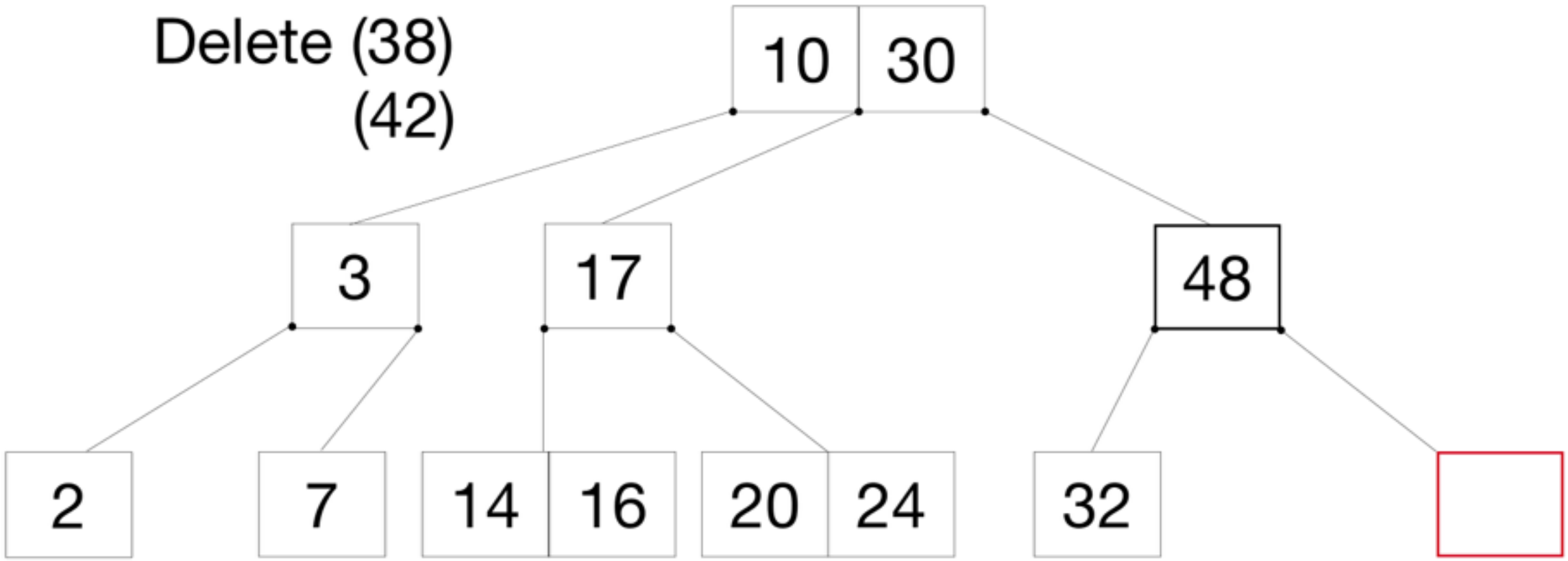
Delete (38)
(42)



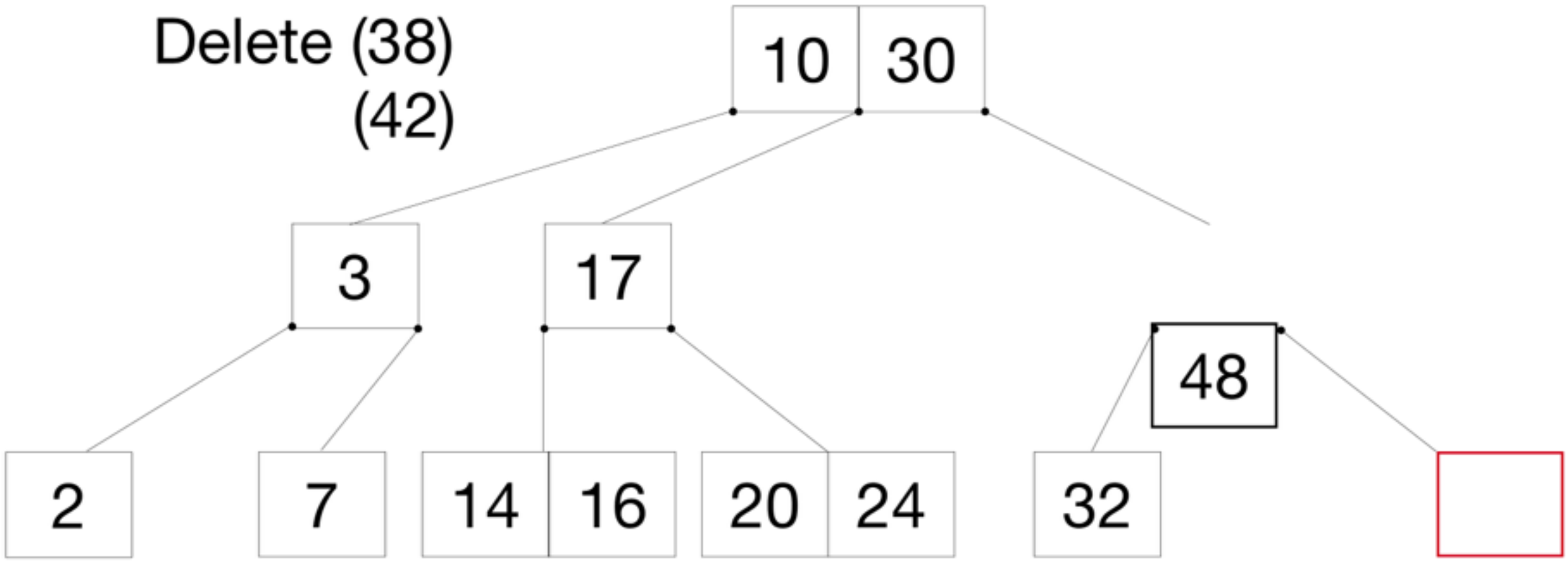
Delete (38)
(42)



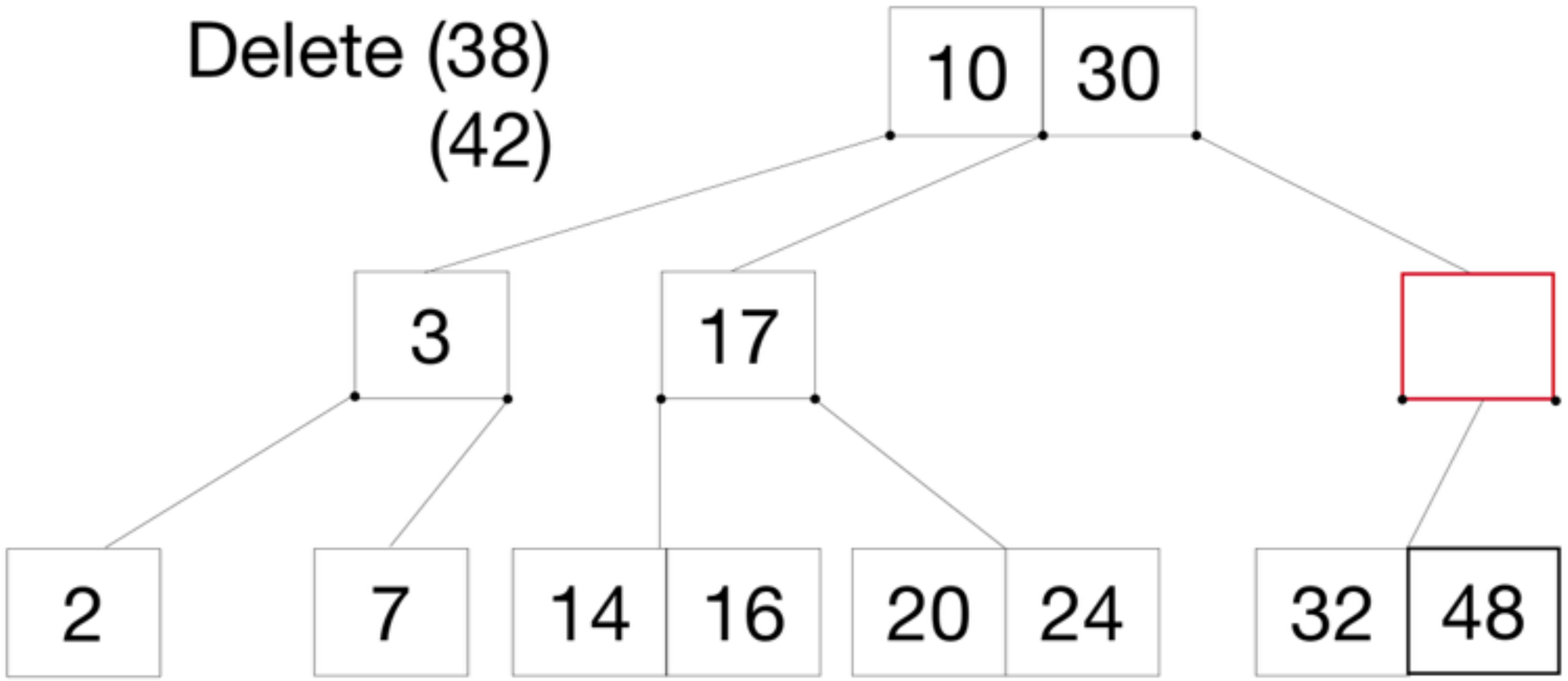
Delete (38)
(42)



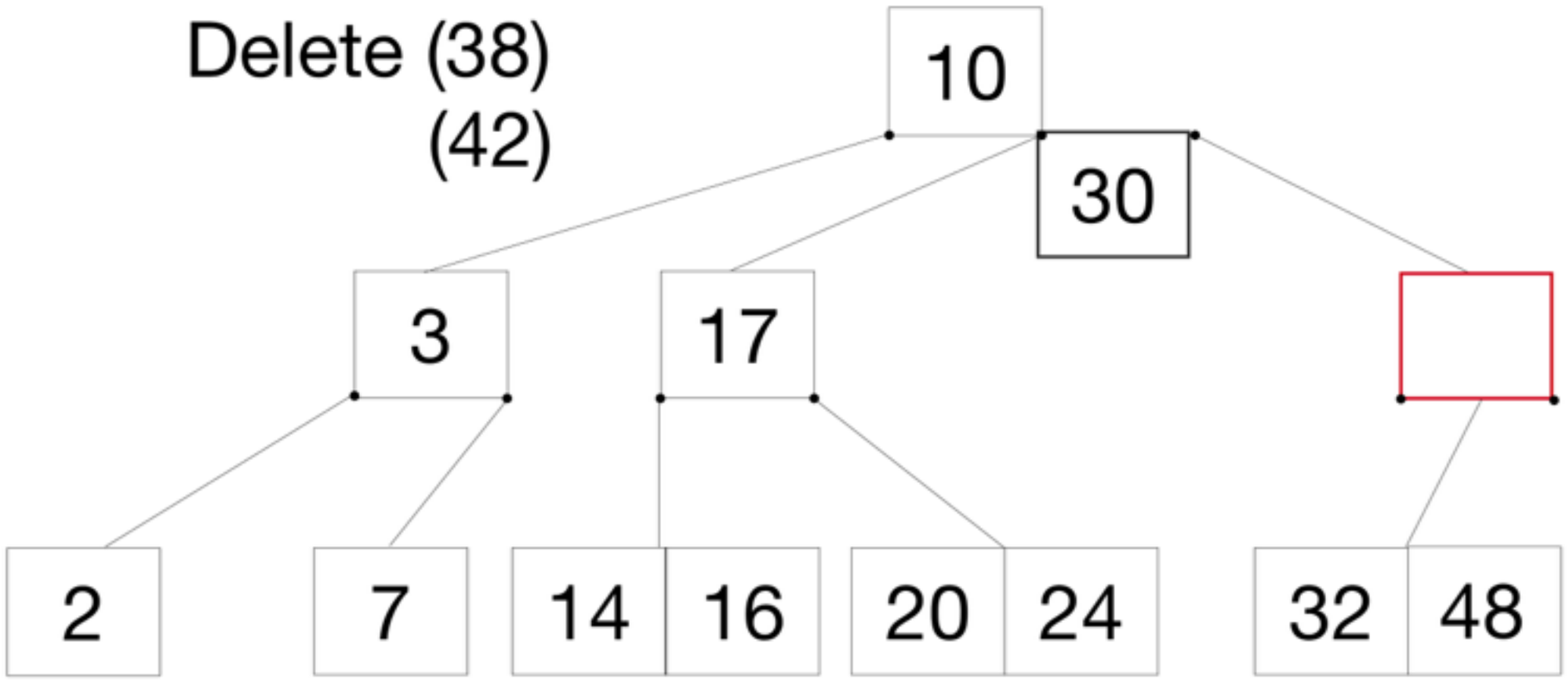
Delete (38)
(42)



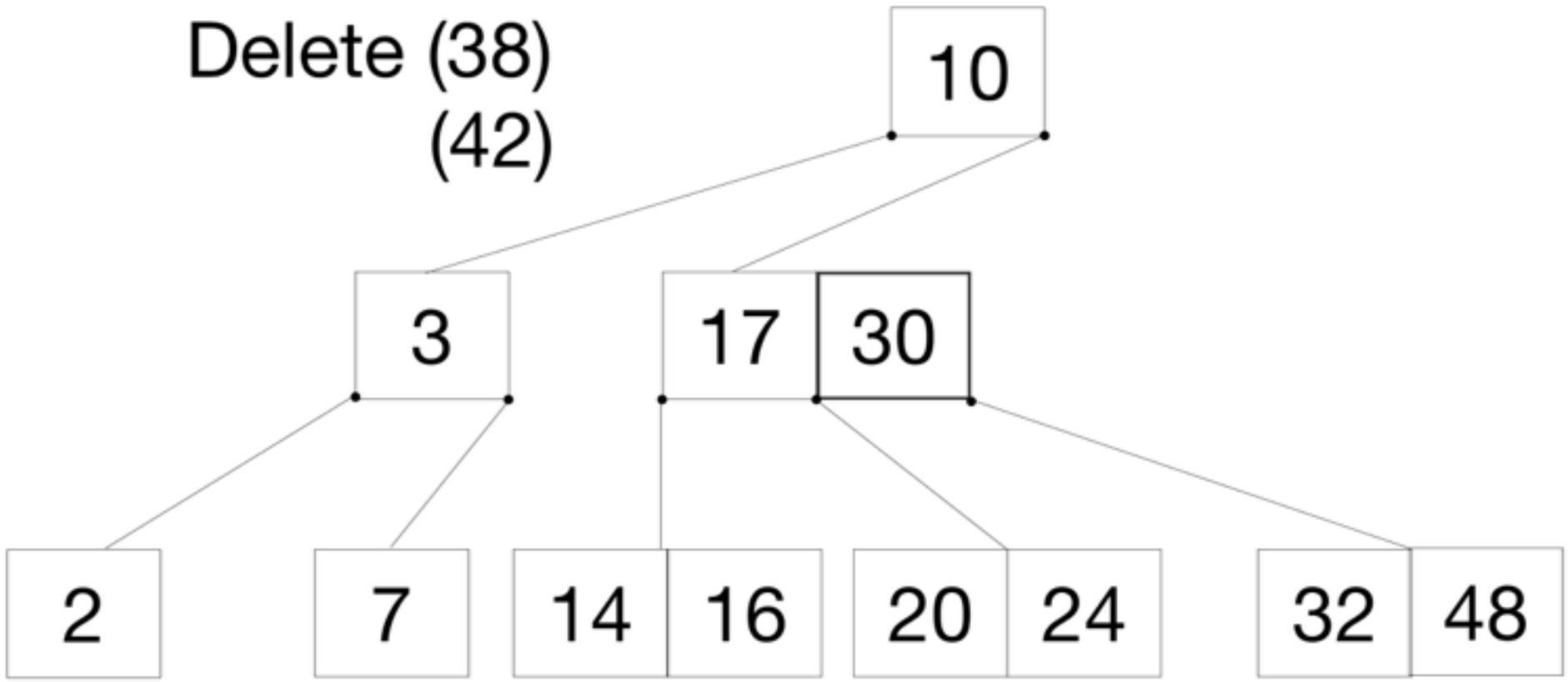
Delete (38)
(42)

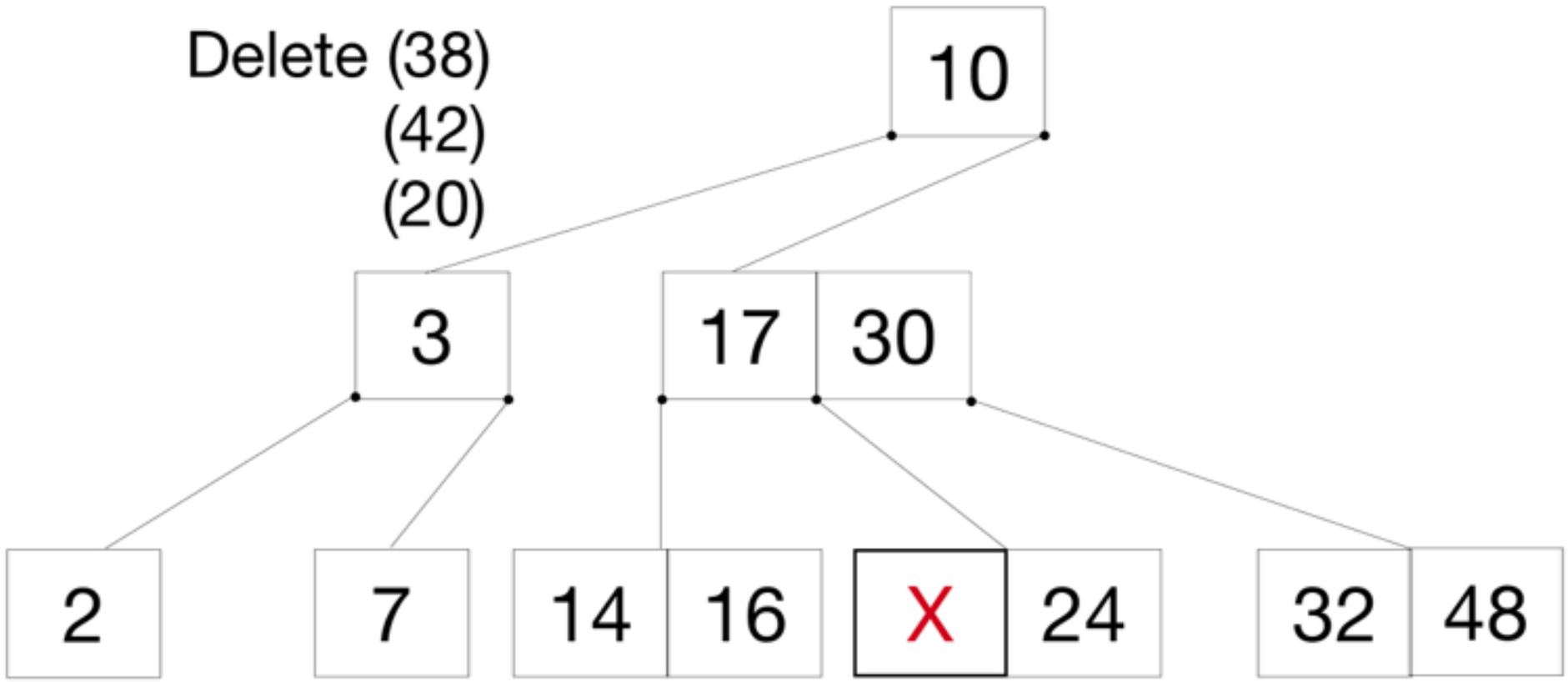


Delete (38)
(42)

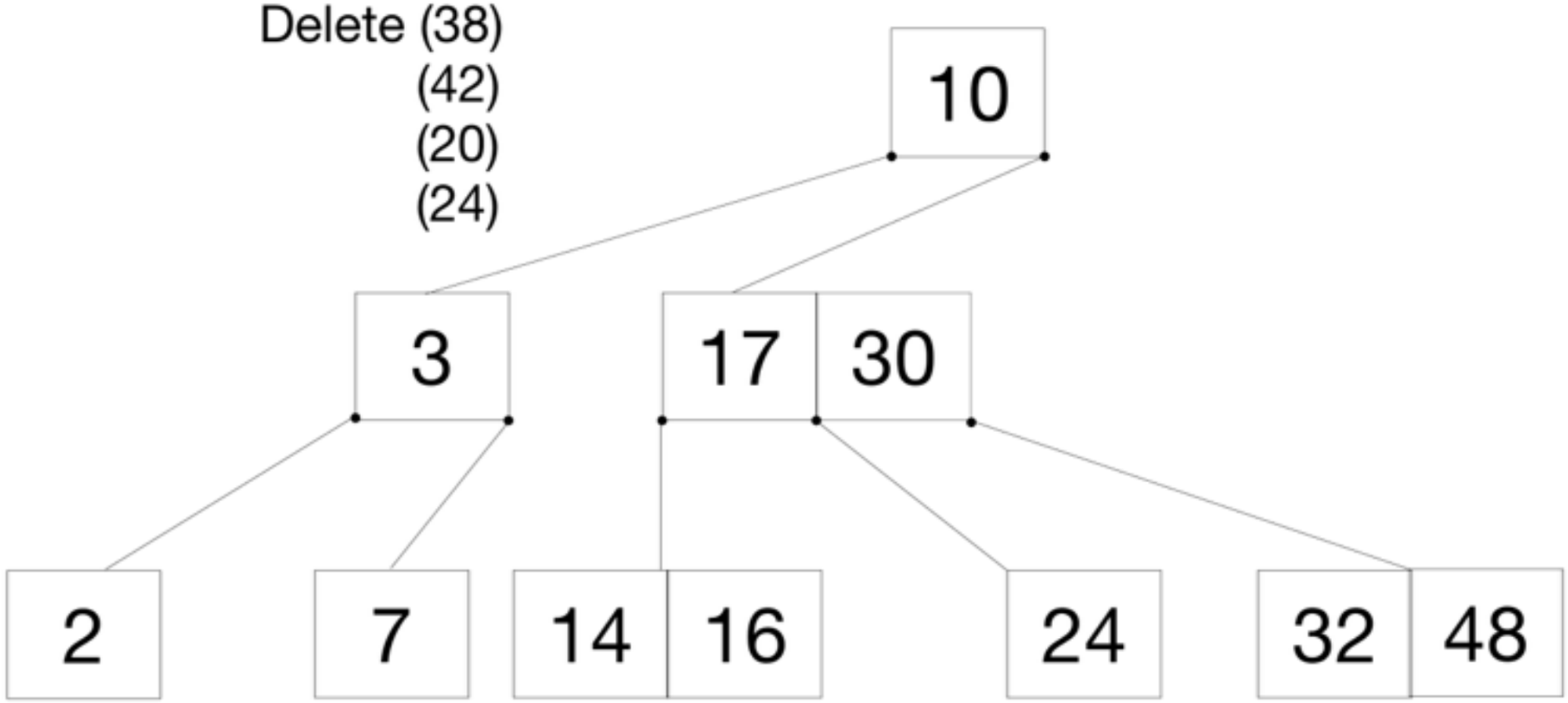


Delete (38)
(42)

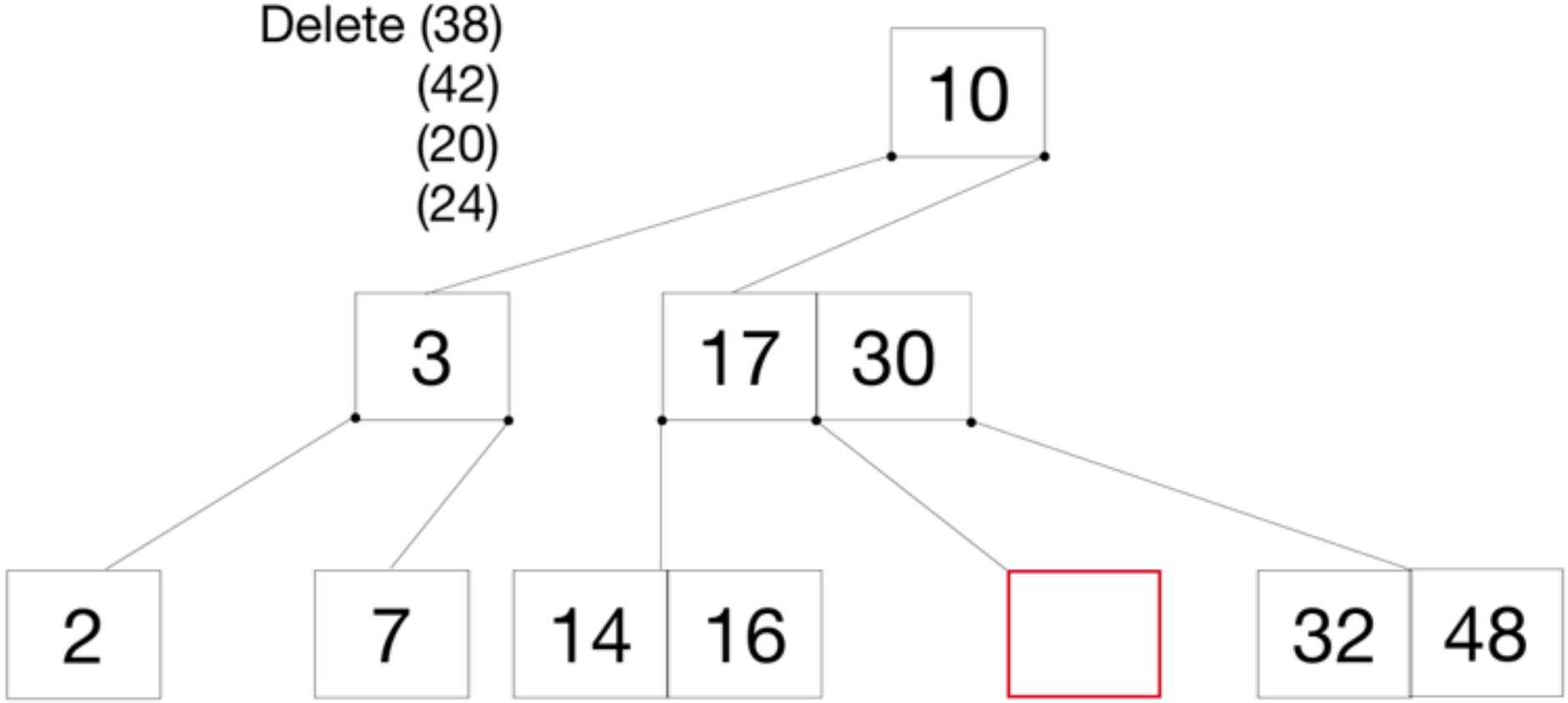



Delete (38)
(42)
(20)
10
3
17
30
2
7
14
16
X
24
32
48

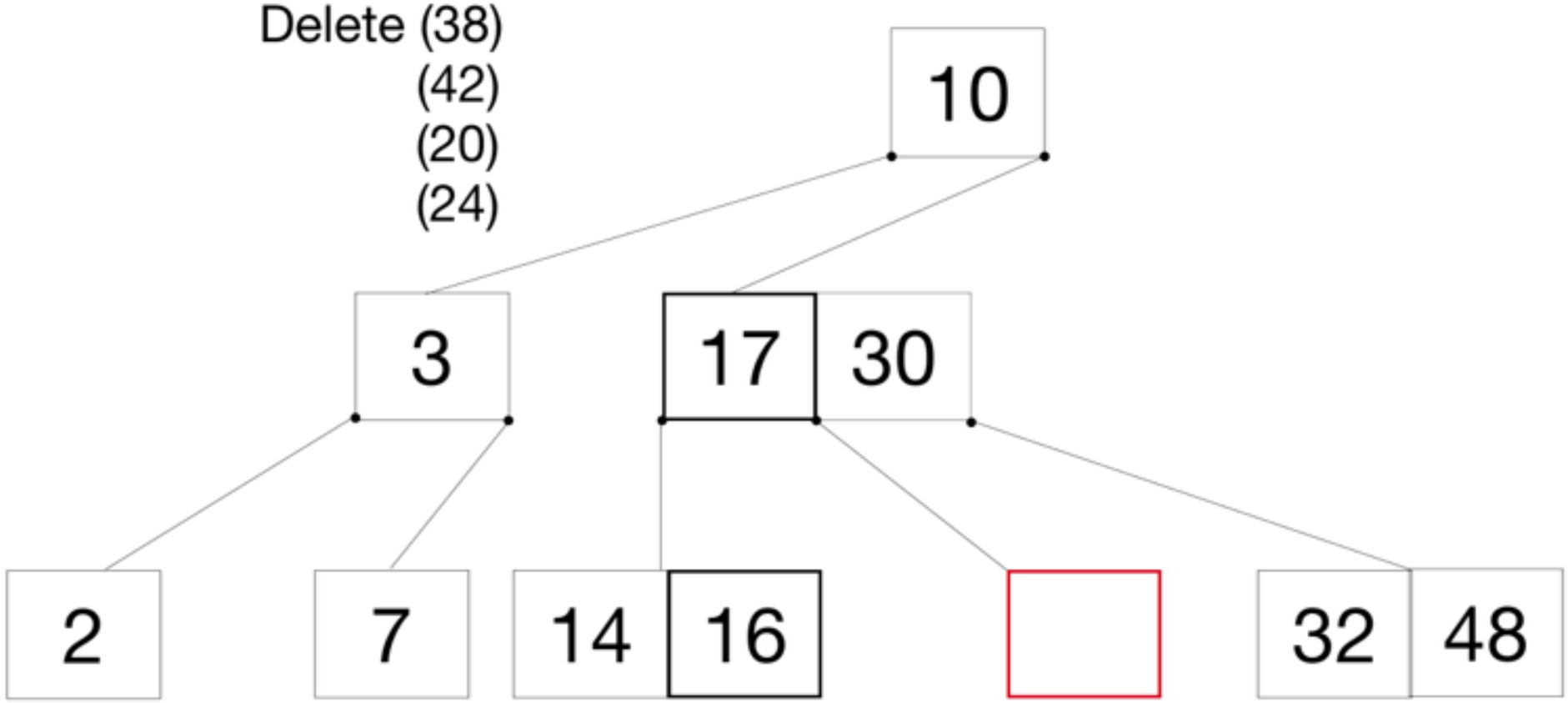
Delete (38)
(42)
(20)
(24)

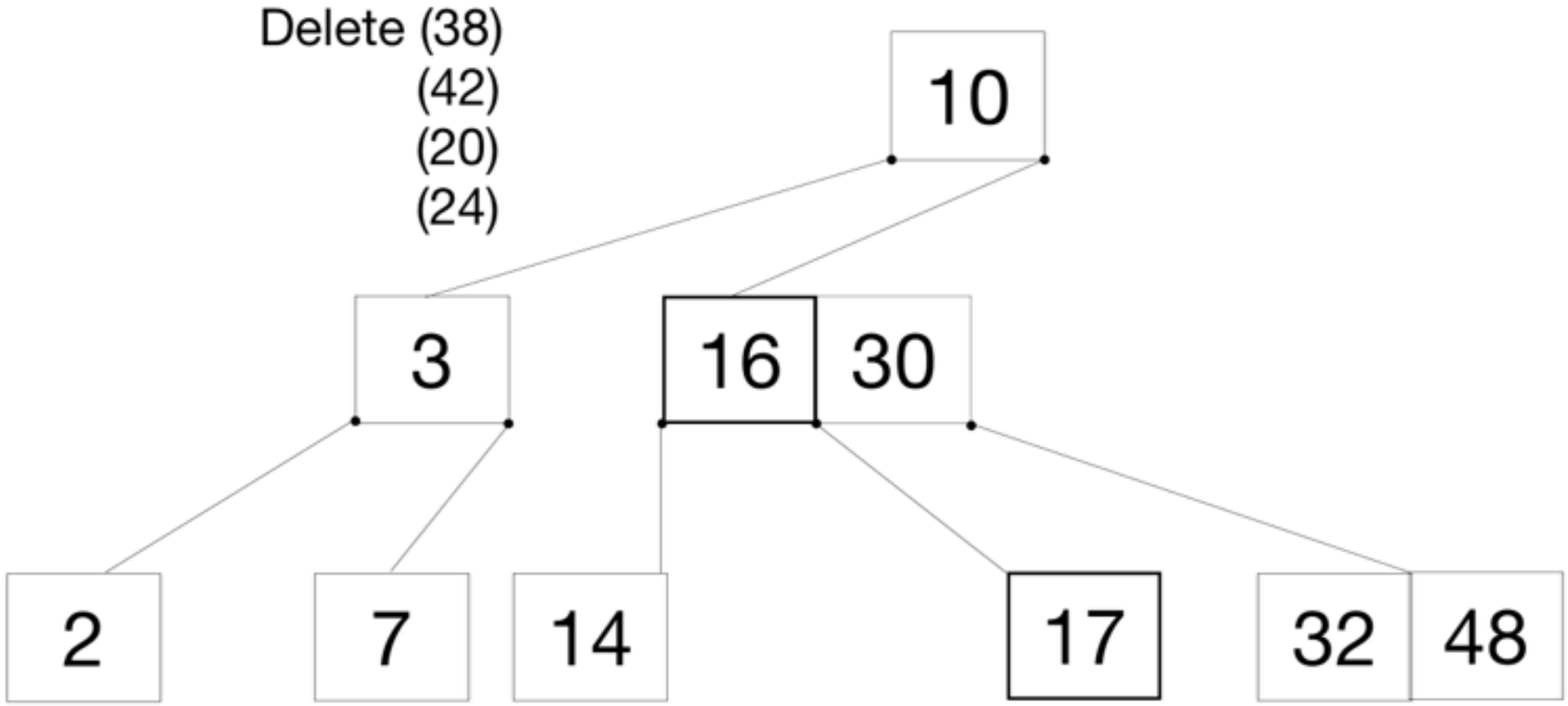


Delete (38)
(42)
(20)
(24)



Delete (38)
(42)
(20)
(24)




Delete (38)
(42)
(20)
(24)
10
3
16
30
2
7
14
17
32
48

summary

purpose : reduce the number of disk accesses.

when : storing HUGE amounts of data

looks like : short and wide

operation time : same as BST

resources

https://www.youtube.com/watch?v=TOb1tuEZ2X4

http://www.geeksforgeeks.org/b-tree-set-1-introduction-2/

https://www.cs.usfca.edu/~galles/visualization/BTree.html

http://www.cs.yale.edu/homes/aspnes/pinewiki/BTrees.html

why ‘B’-Tree?

Bayer? 👨💻

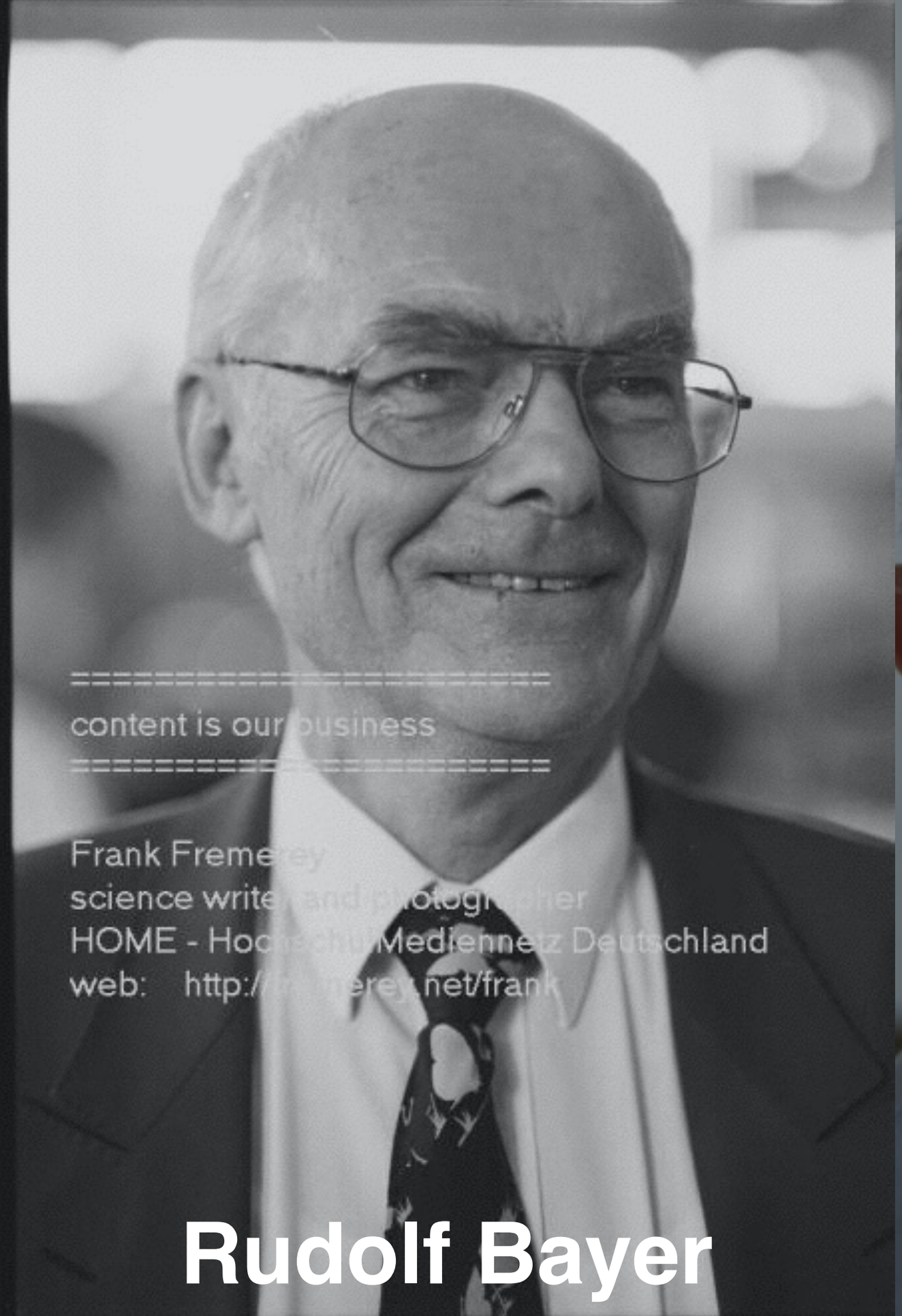
Boeing? 

why ‘B’-Tree?

Broad? 🗿

Bushy? 🌳

Balanced? t



Rudolf Bayer

Ed McCreight

“What really lives to say is: the more you think about what the B in B-trees means, the better you understand B-trees.”

— Dr. Rudolf Bayer, co-inventor of the B-tree