B+ & 2-3 Trees

Kuan-Yu Chen (陳冠宇)

2018/10/29 @ TR-212, NTUST

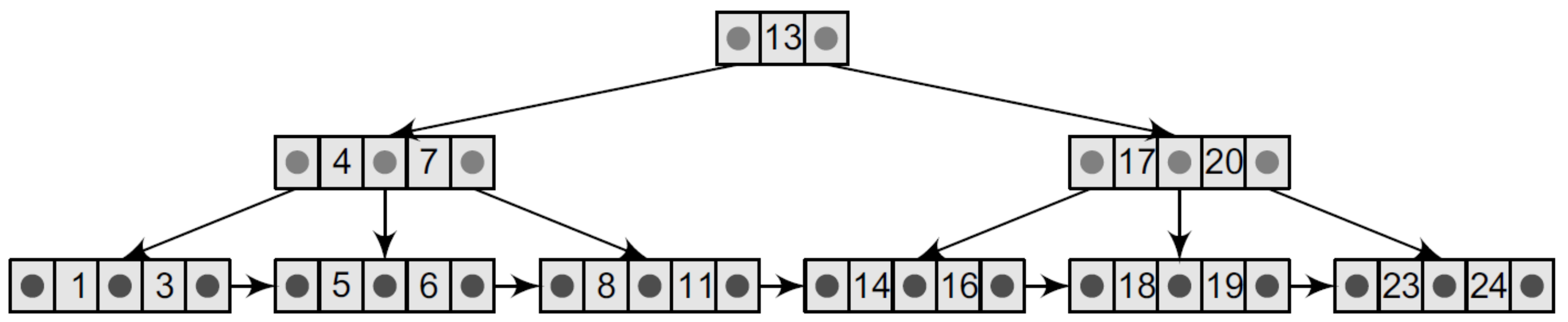
Review

- Multi-way Search Trees
  - M is called the degree of the tree
  - If M=2, each node in the M-way search tree has one value and two sub-trees
    - Binary Search Tree!

- B Trees
  - Every node in the B tree has at most (maximum) m children
  - Every node in the B tree except the root node and leaf nodes has at least (minimum) $\left\lceil \frac{m}{2} \right\rceil$ children
    - Degree=4, at least 2 children, at least 1 key
    - Degree=5, at least 3 children, at least 2 key
  - The root node has at least two children
  - All leaf nodes are at the same level

B+ Trees

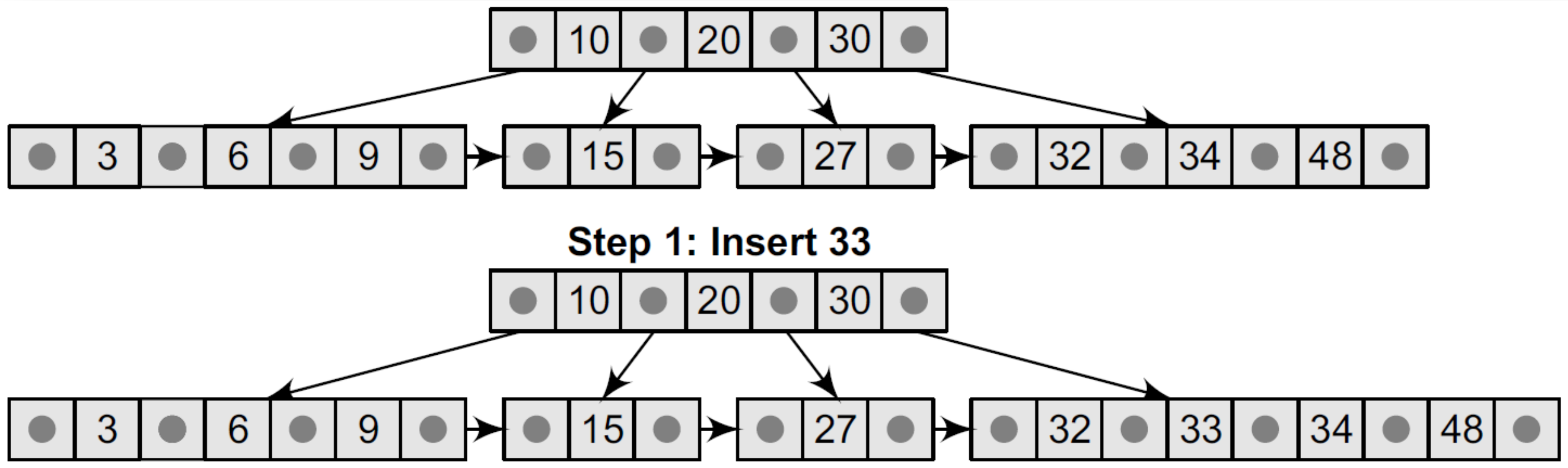
- A B+ tree is a variant of a B tree which stores sorted data in a way that allows for efficient insertion, retrieval, and removal of records, each of which is identified by a key
  - A B tree can store both keys and records in its interior nodes
  - A B+ tree stores all the records at the leaf level of the tree and keys are stored in the interior nodes
    - Typically, B+ trees are used to store large amounts of data that cannot be stored in the main memory
    - The leaf nodes of a B+ tree are often linked to one another in a linked list
    - All of the internal nodes are called index nodes or i-nodes



B+ Trees – Insertion.

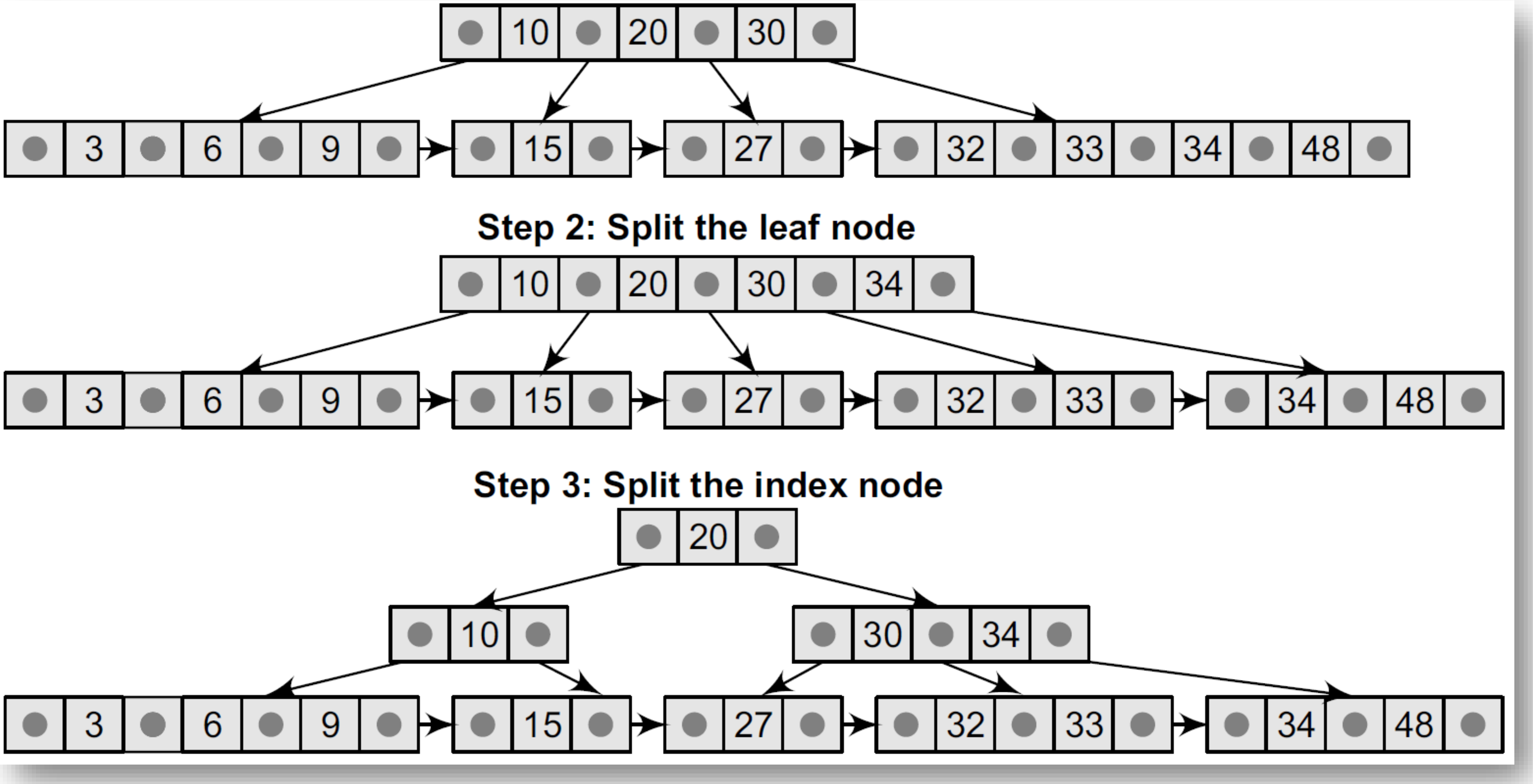
- For inserting a new element in a B+ tree
  - A new element is simply added in the leaf node if there is space for it
  - If the data node in the tree is full, then that node is split into two nodes

- For a given B+ tree of order 4, please insert 33 in the tree



B+ Trees – Insertion..

- For a given B+ tree of order 4, please insert 33 in the tree

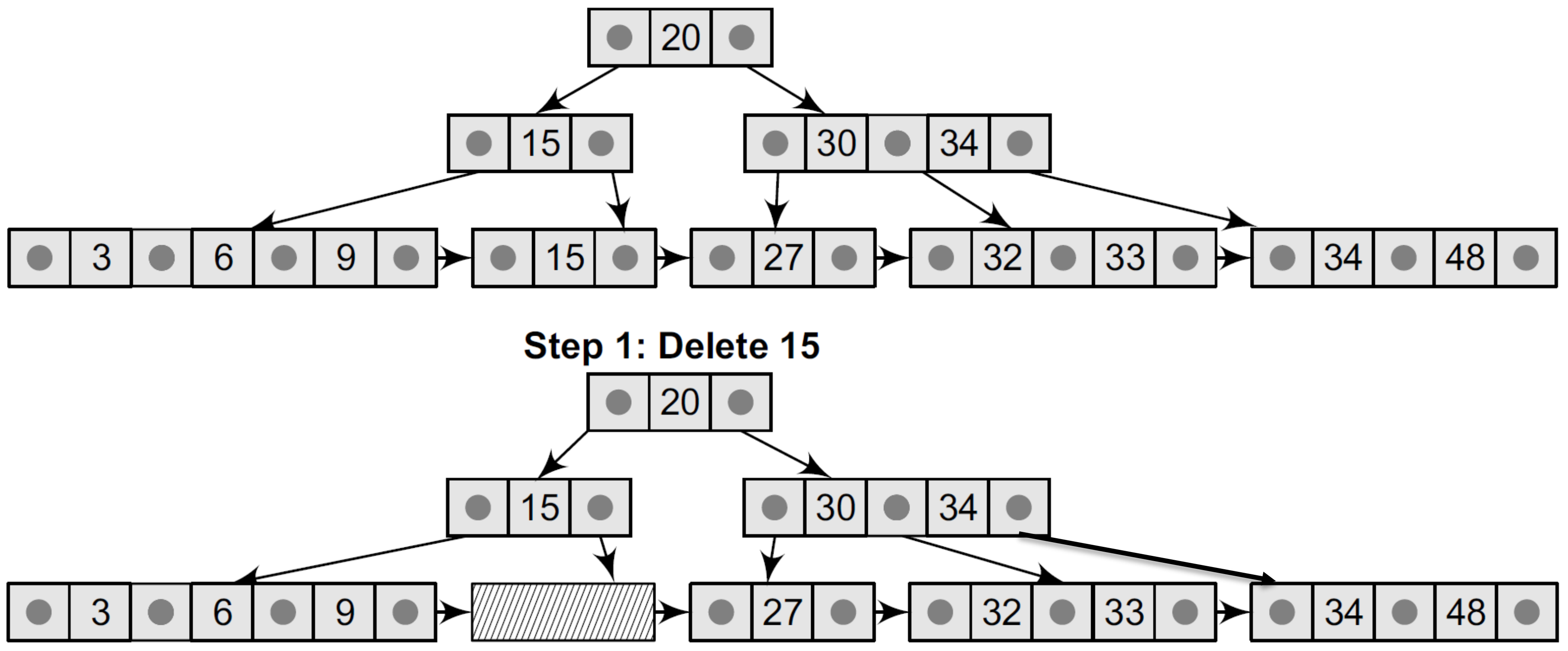


B+ Trees – Deletion.

- For a B+ tree, deletion is always done from a leaf node
  1. Delete the key and data from the leaves
  2. If the **leaf node underflows**, merge that node with the sibling and **delete** the key in between them
  3. If the **index node underflows**, merge that node with the sibling and **move down** the key in between them.

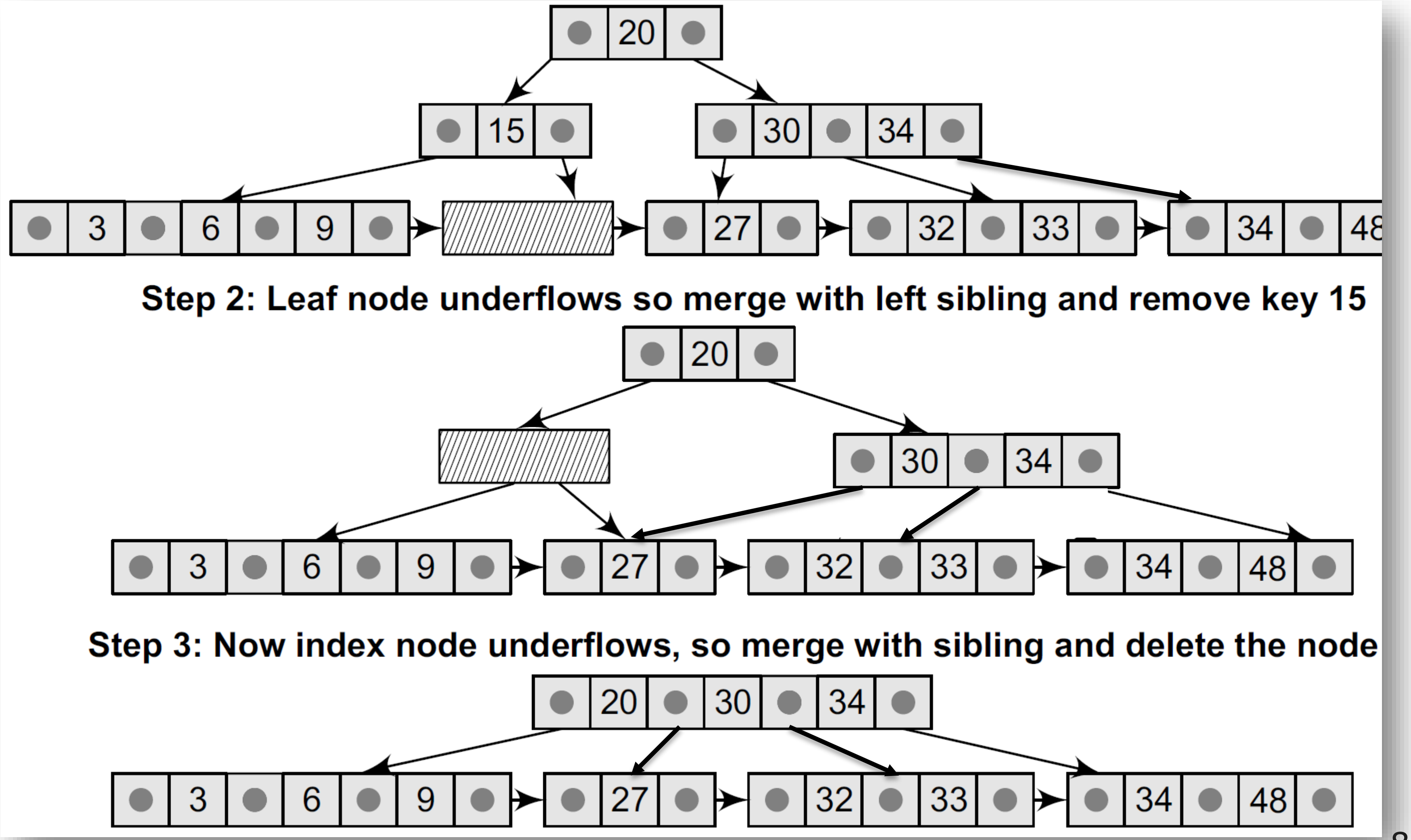
B+ Trees – Deletion..

- For a B+ tree of order 4, please delete node 15 from the tree



B+ Trees – Deletion…

- For a B+ tree of order 4, please delete node 15 from the tree

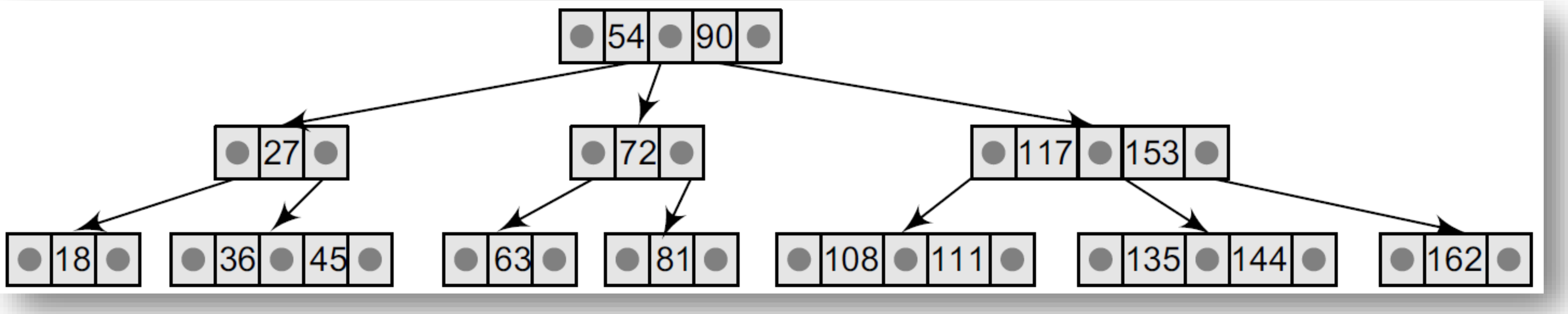


Step 2: Leaf node underflows so merge with left sibling and remove key 15

Step 3: Now index node underflows, so merge with sibling and delete the node

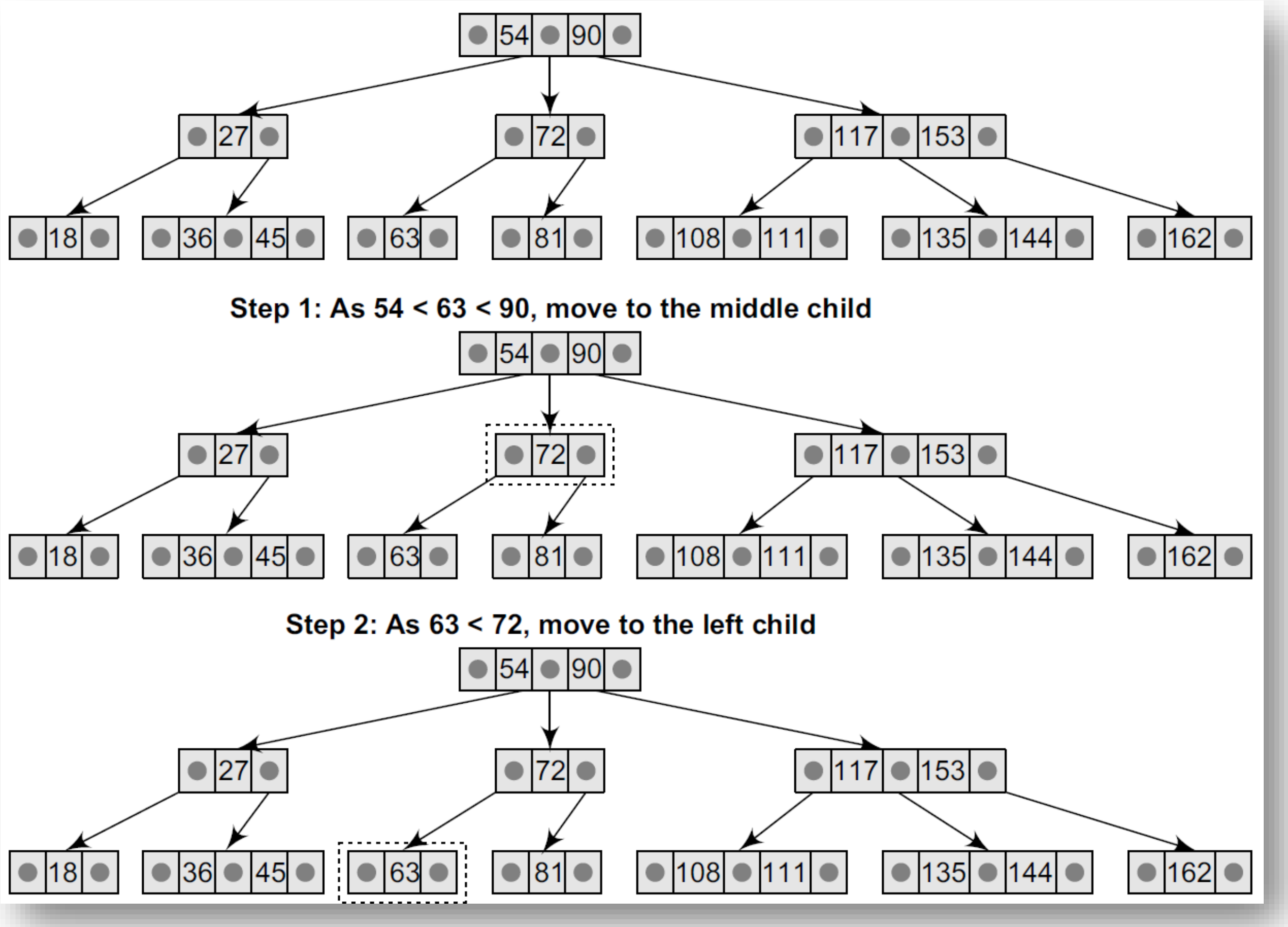
2-3 Trees

- In a 2-3 tree, each interior node has either two or three children
  - Nodes with two children are called 2-nodes
    - The 2-nodes have one data value and two children
  - Nodes with three children are called 3-nodes
    - The 3-nodes have two data values and three children
  - All the leaf nodes are at the same level



2-3 Trees – Searching

- The search operation is used to determine whether a data value is present in a 2-3 tree
  - Search for 63

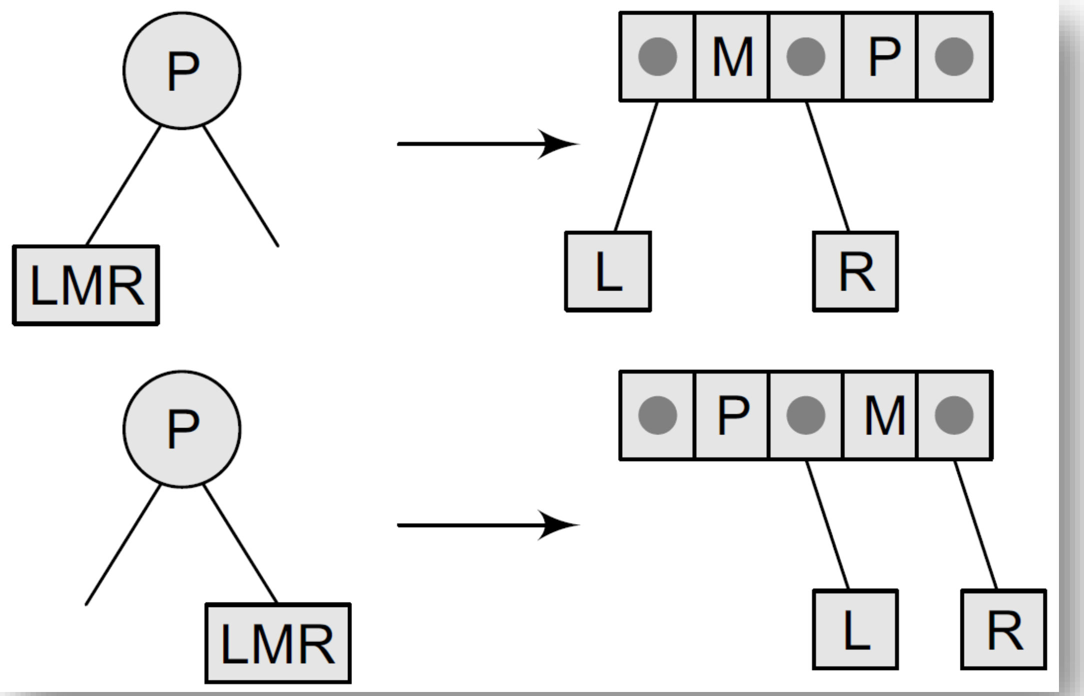


Step 1: As 54 < 63 < 90, move to the middle child

Step 2: As 63 < 72, move to the left child

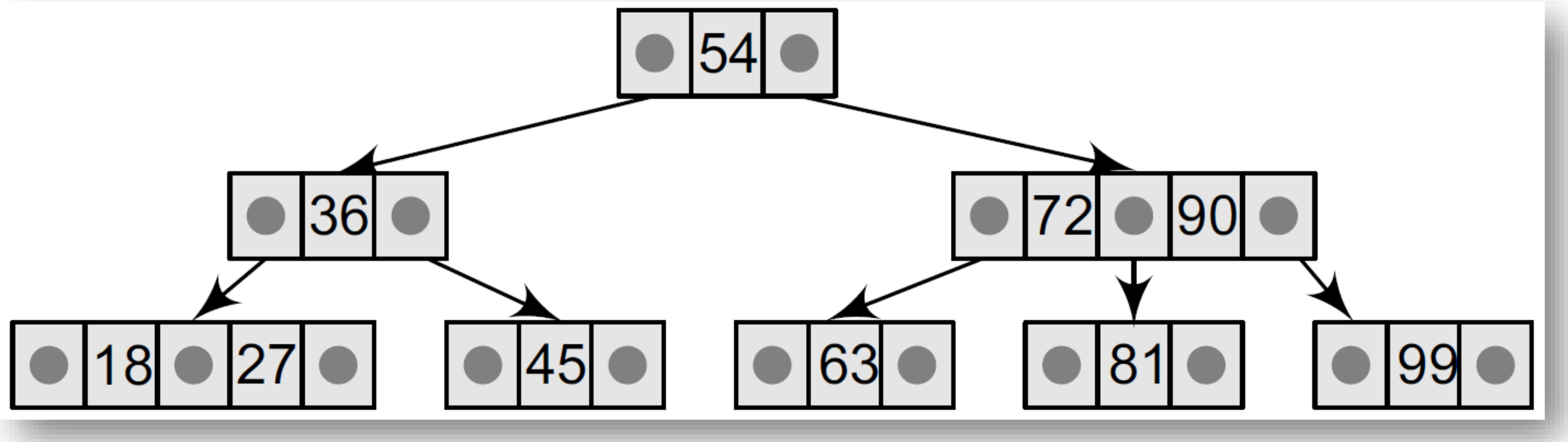
2-3 Trees – Insertion.

- To insert a new value in the 2-3 tree, an appropriate position of the value is located in one of the leaf nodes
  - If after insertion of the new value, the properties of the 2-3 tree do not get violated then insertion is over
  - If any property is violated then the violating node must be split
    - A node is split when it has three data values and four children

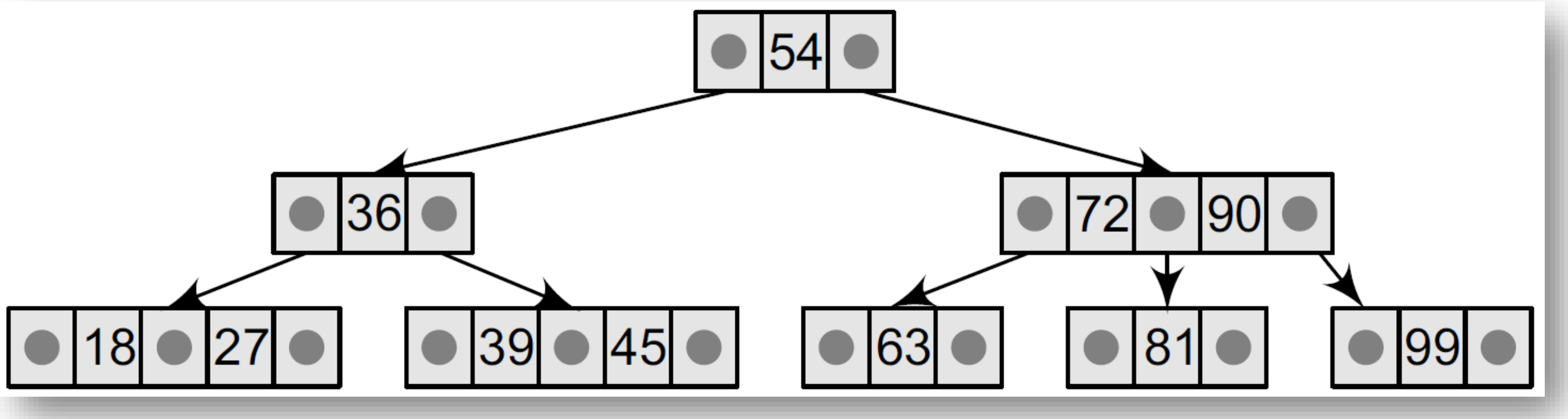


2-3 Trees – Insertion..

- Given a 2-3 tree, please insert the data values into the tree 39, 37, 42, 47

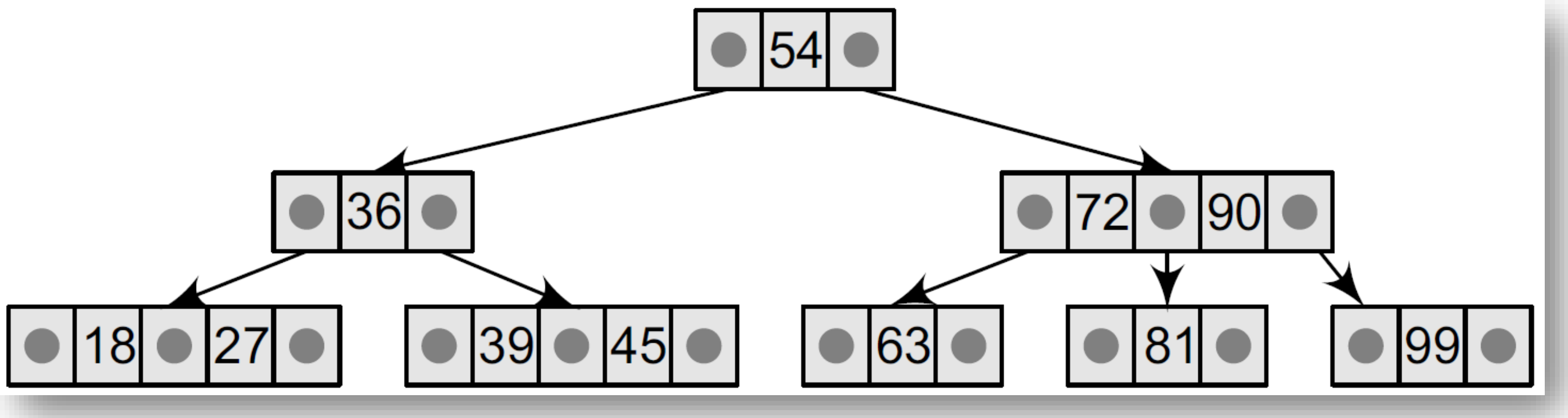


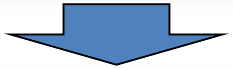
Insert 39

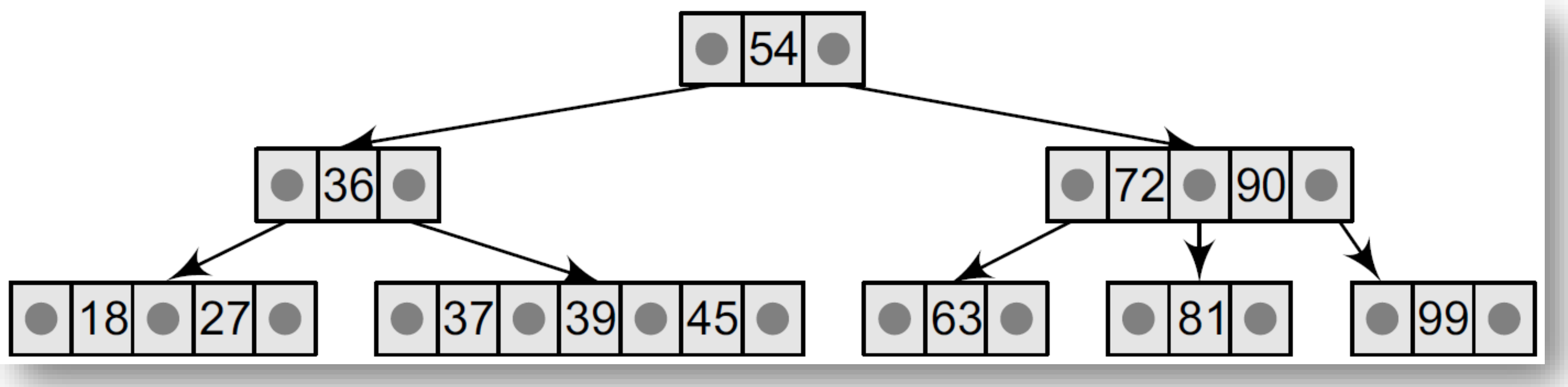


2-3 Trees – Insertion…

- Given a 2-3 tree, please insert the data values into the tree 39, 37, 42, 47

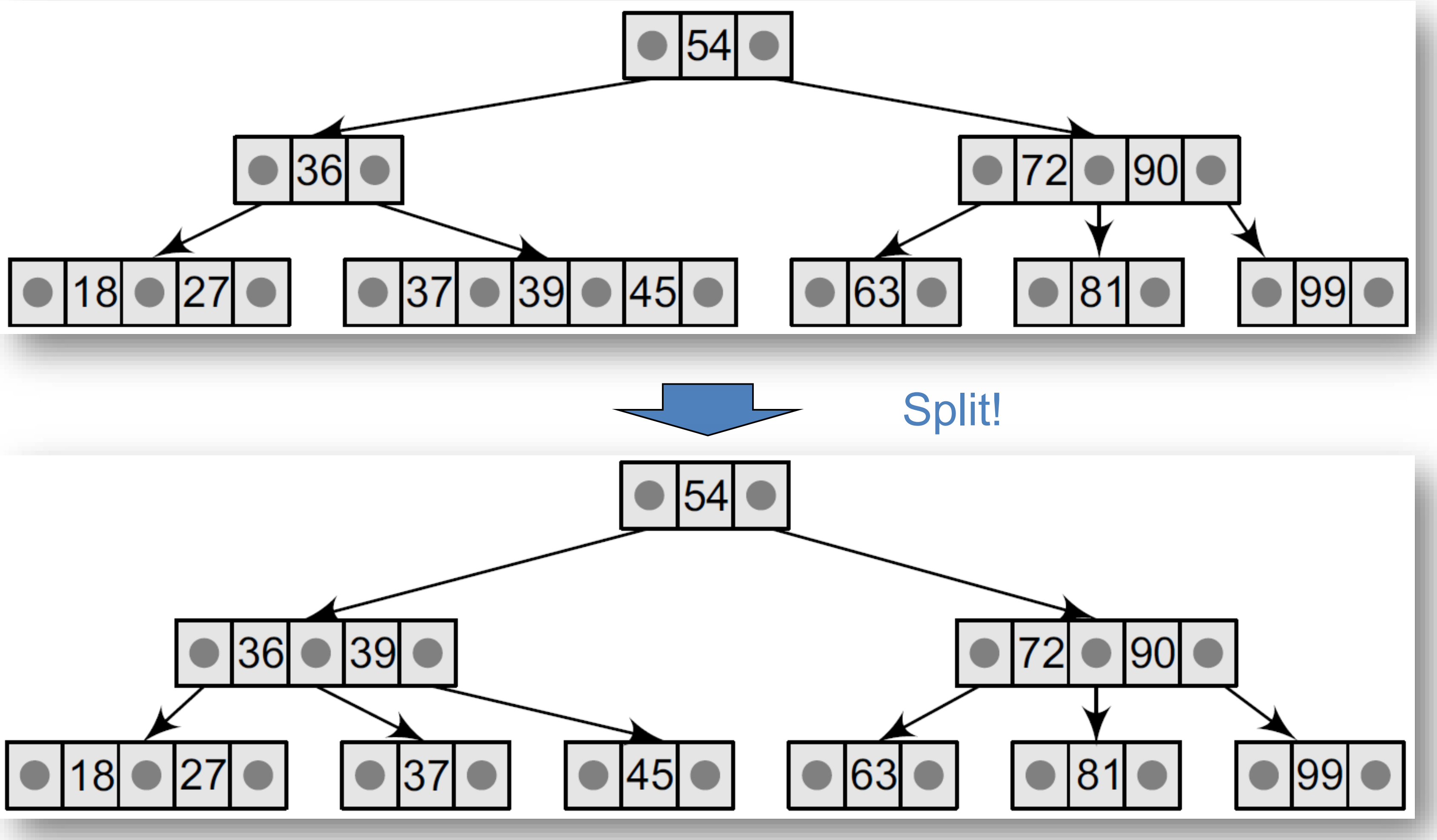



Insert 37



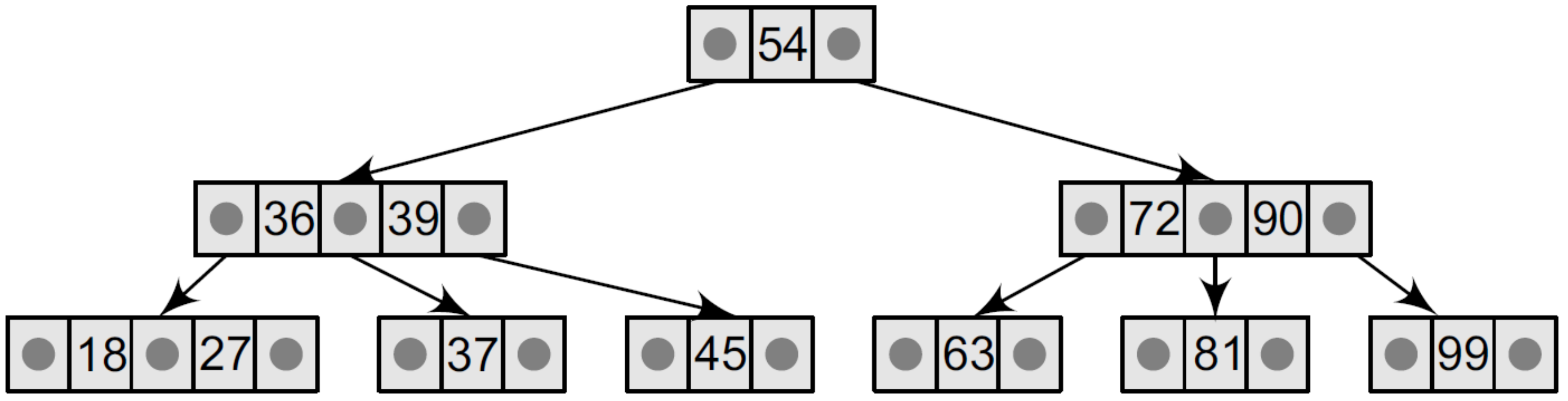
2-3 Trees – Insertion….

- Given a 2-3 tree, please insert the data values into the tree 39, 37, 42, 47

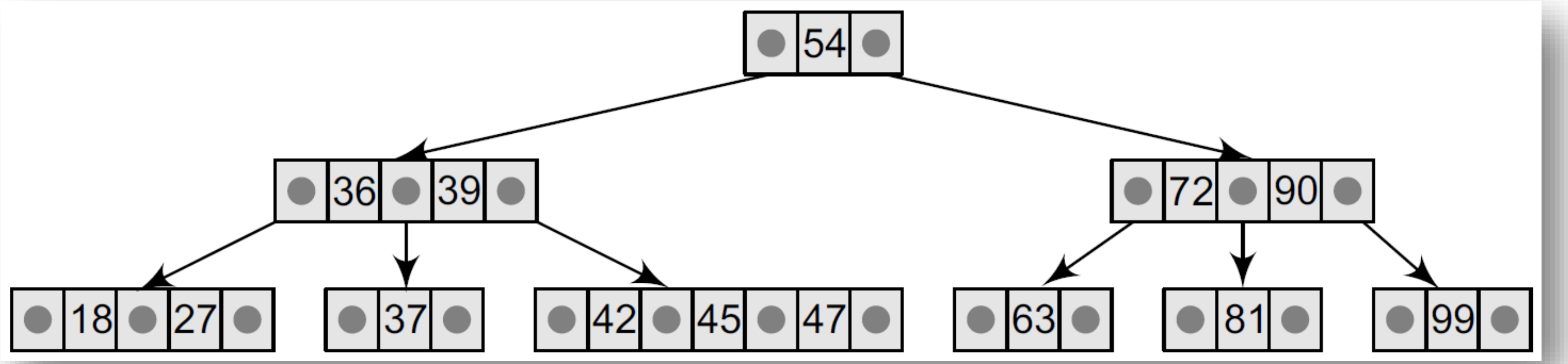


2-3 Trees – Insertion…..

- Given a 2-3 tree, please insert the data values into the tree 39, 37, 42, 47

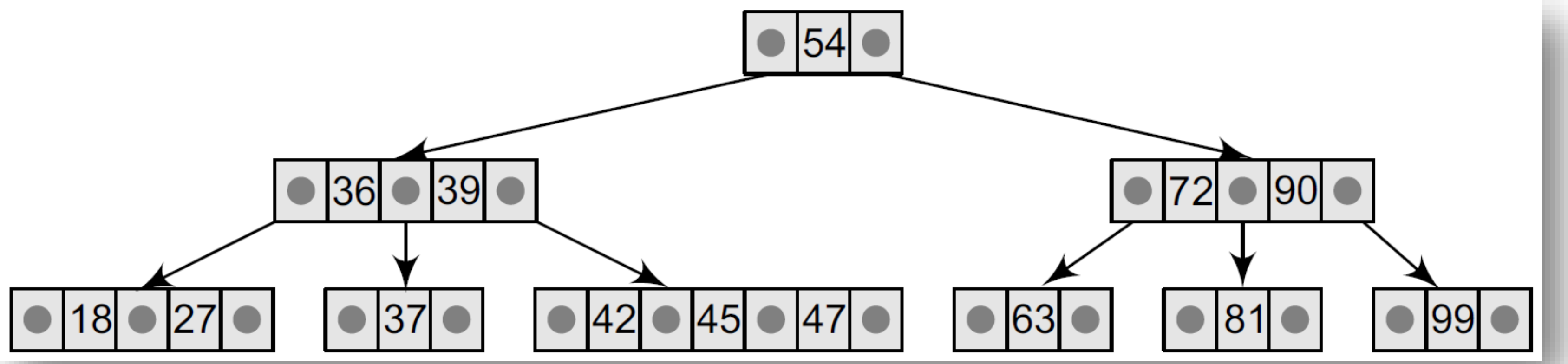


Insert 42 & 47

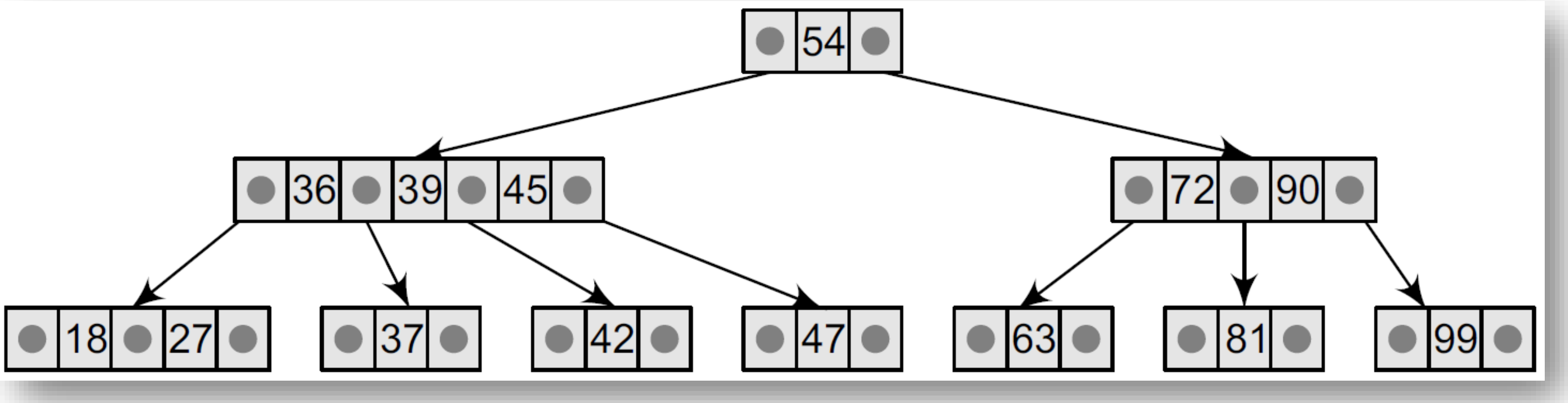


2-3 Trees – Insertion…..

- Given a 2-3 tree, please insert the data values into the tree 39, 37, 42, 47

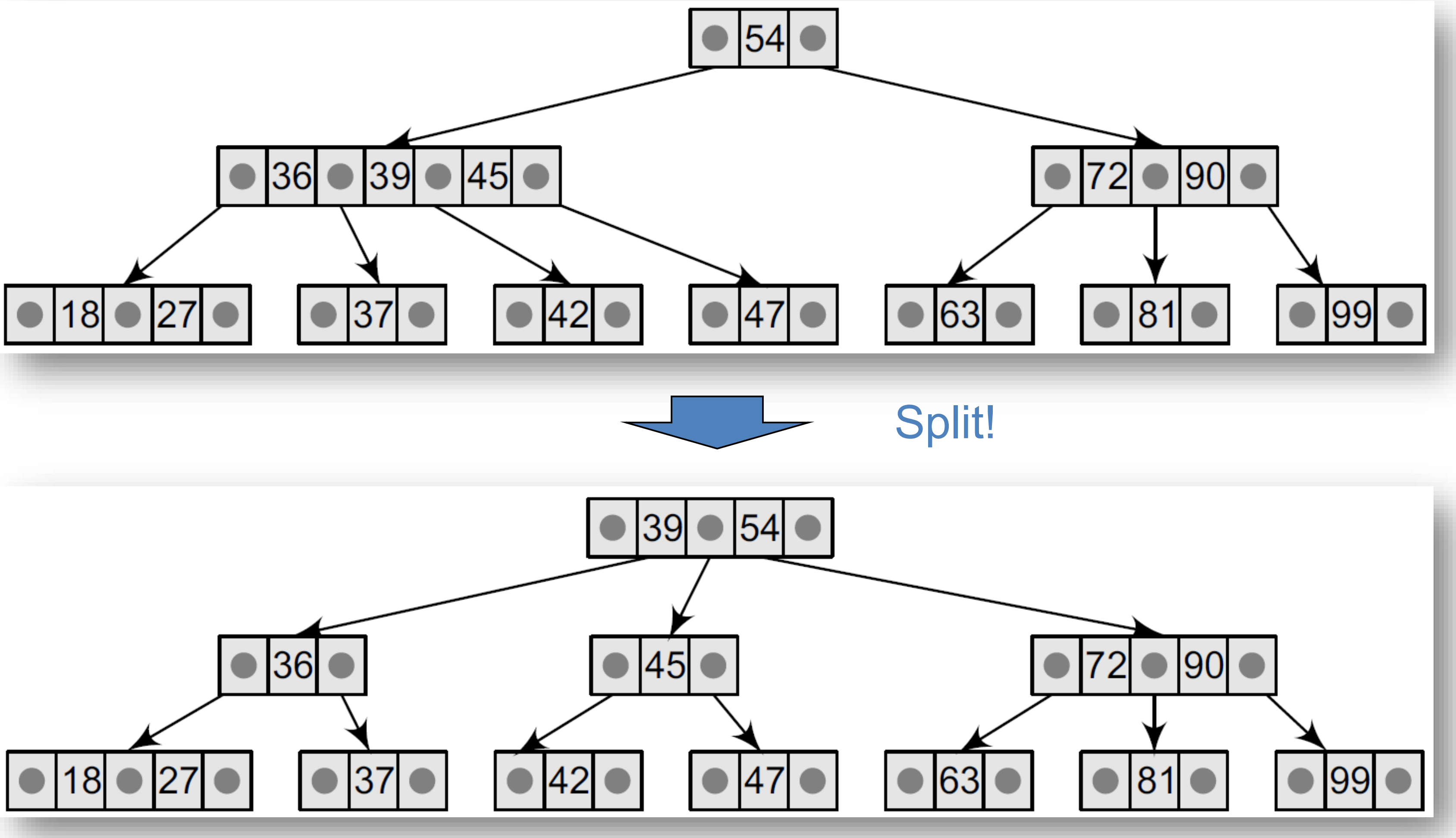


Split!



2-3 Trees – Insertion……

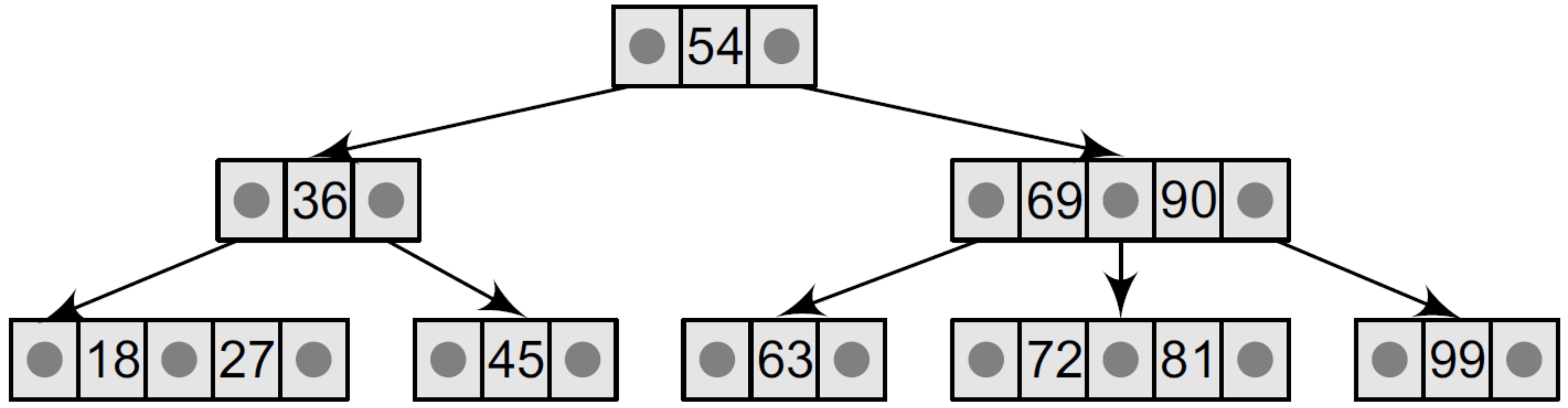
- Given a 2-3 tree, please insert the data values into the tree 39, 37, 42, 47



2-3 Trees – Deletion.

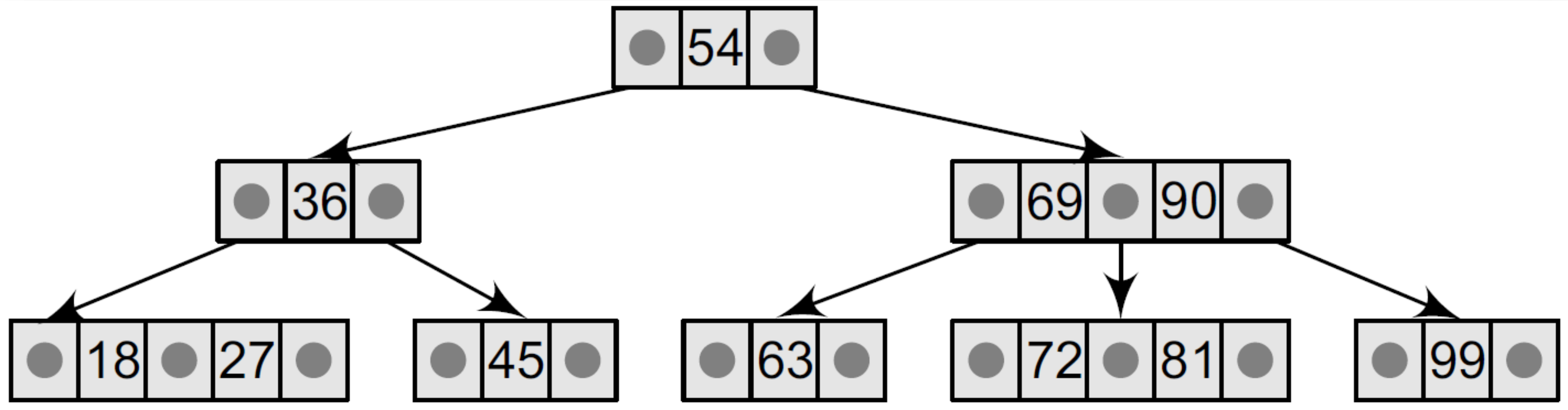
- To delete a value, it is replaced by its in-order successor and then removed
  - If a node becomes empty after deleting a value, it is then merged with another node to restore the property of the tree

- Given a 2-3 tree, please delete the values 69, 72, 99, 81

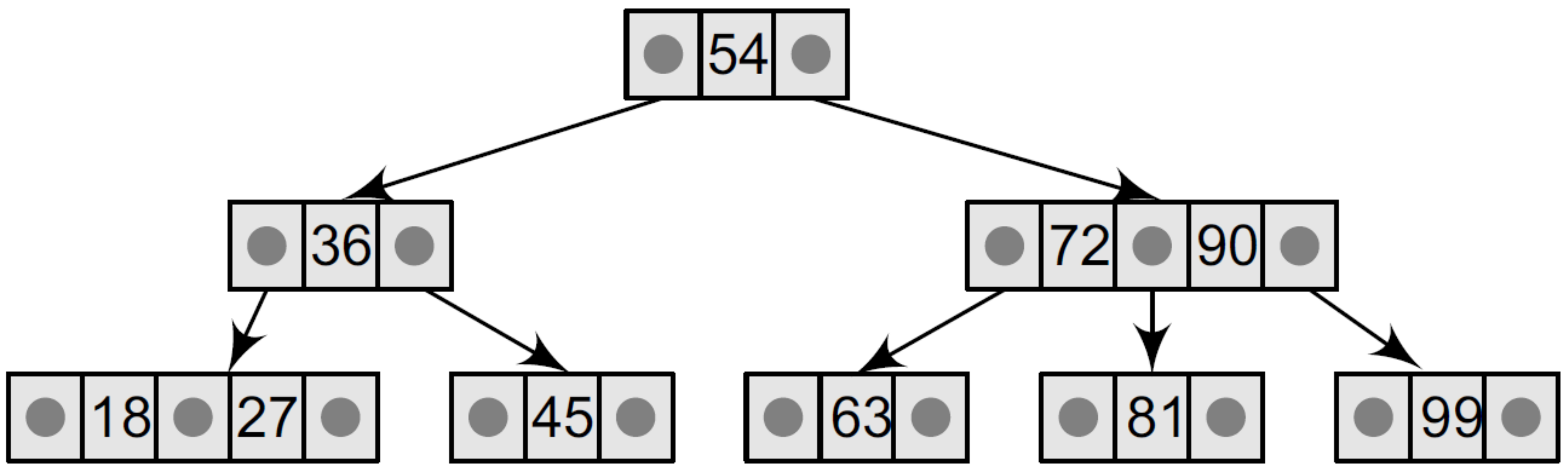


2-3 Trees – Deletion..

- Given a 2-3 tree, please delete the values 69, 72, 99, 81

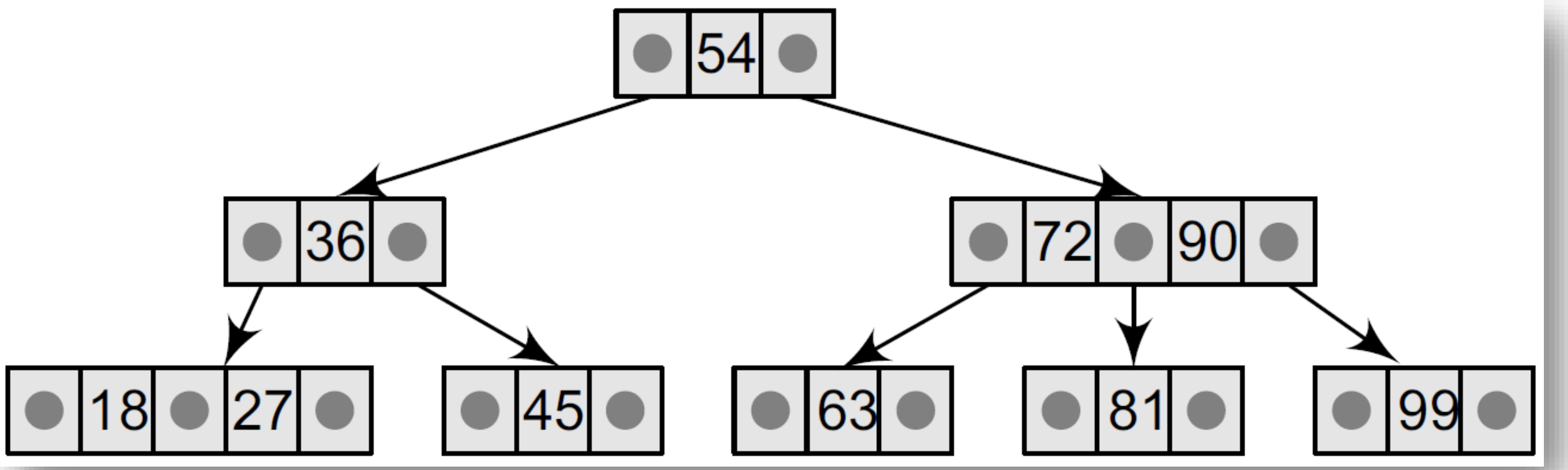


Delete 69

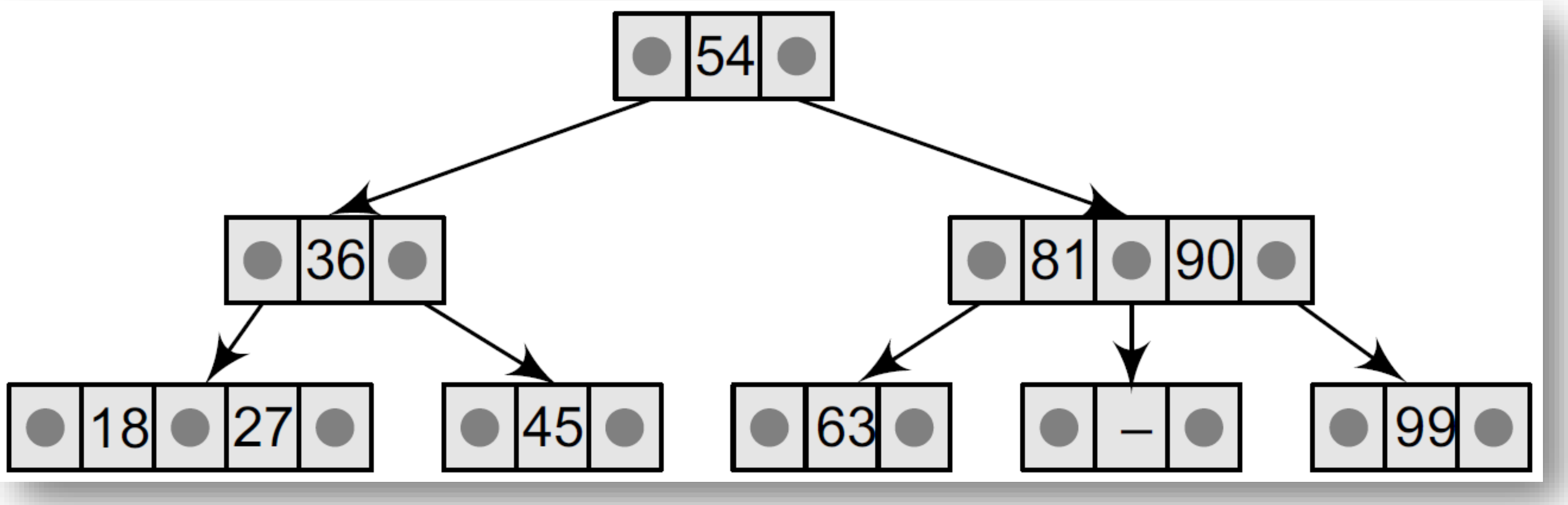


2-3 Trees – Deletion...

- Given a 2-3 tree, please delete the values 69, 72, 99, 81

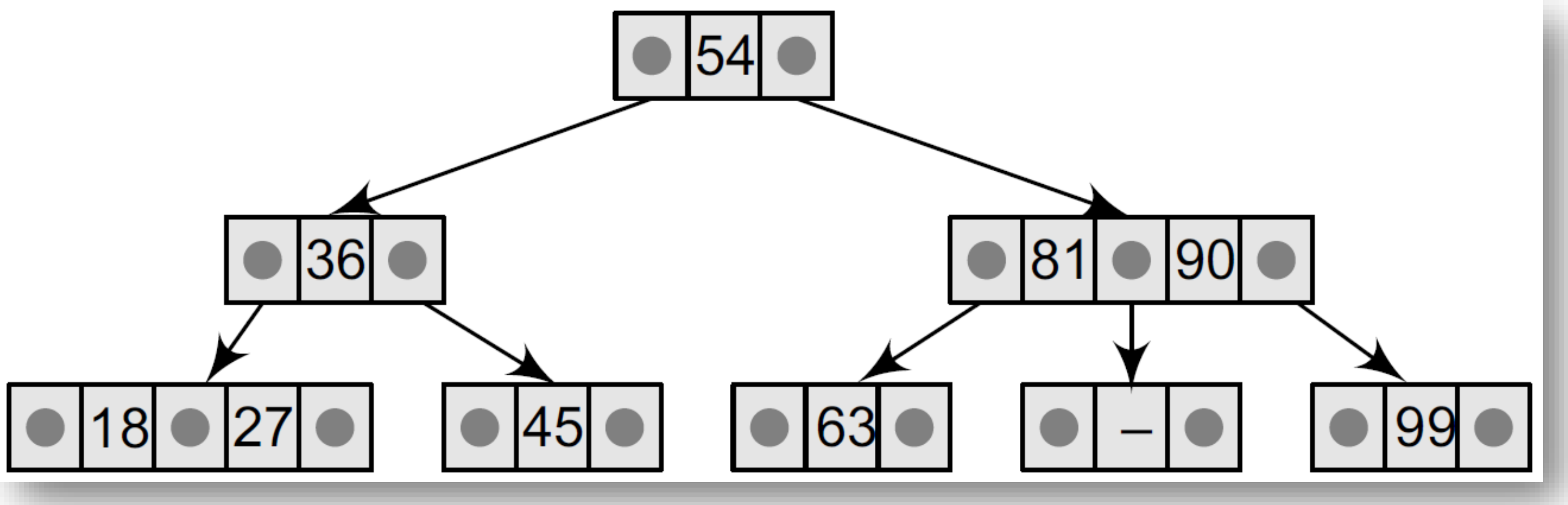


Delete 72

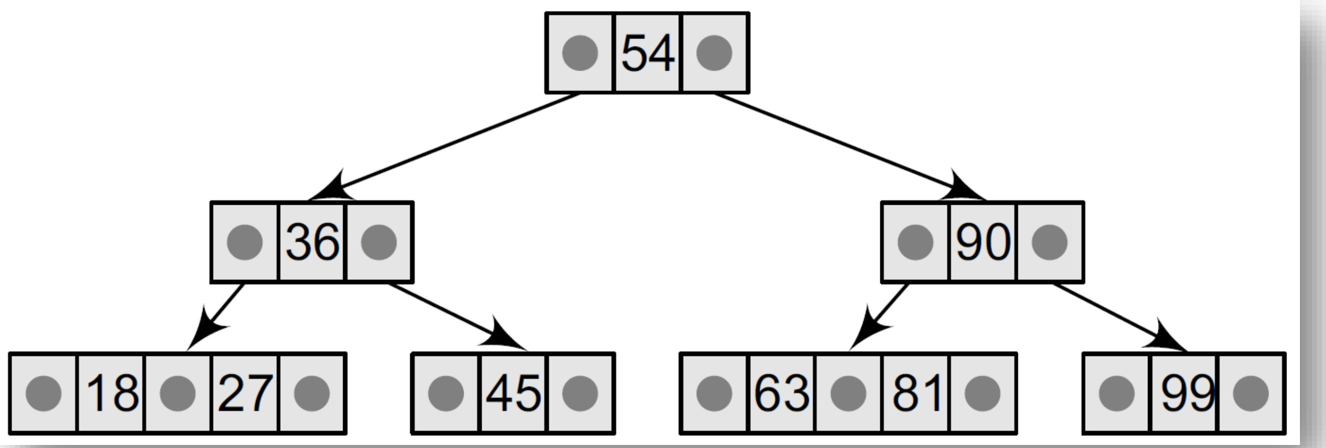


2-3 Trees – Deletion….

- Given a 2-3 tree, please delete the values 69, 72, 99, 81

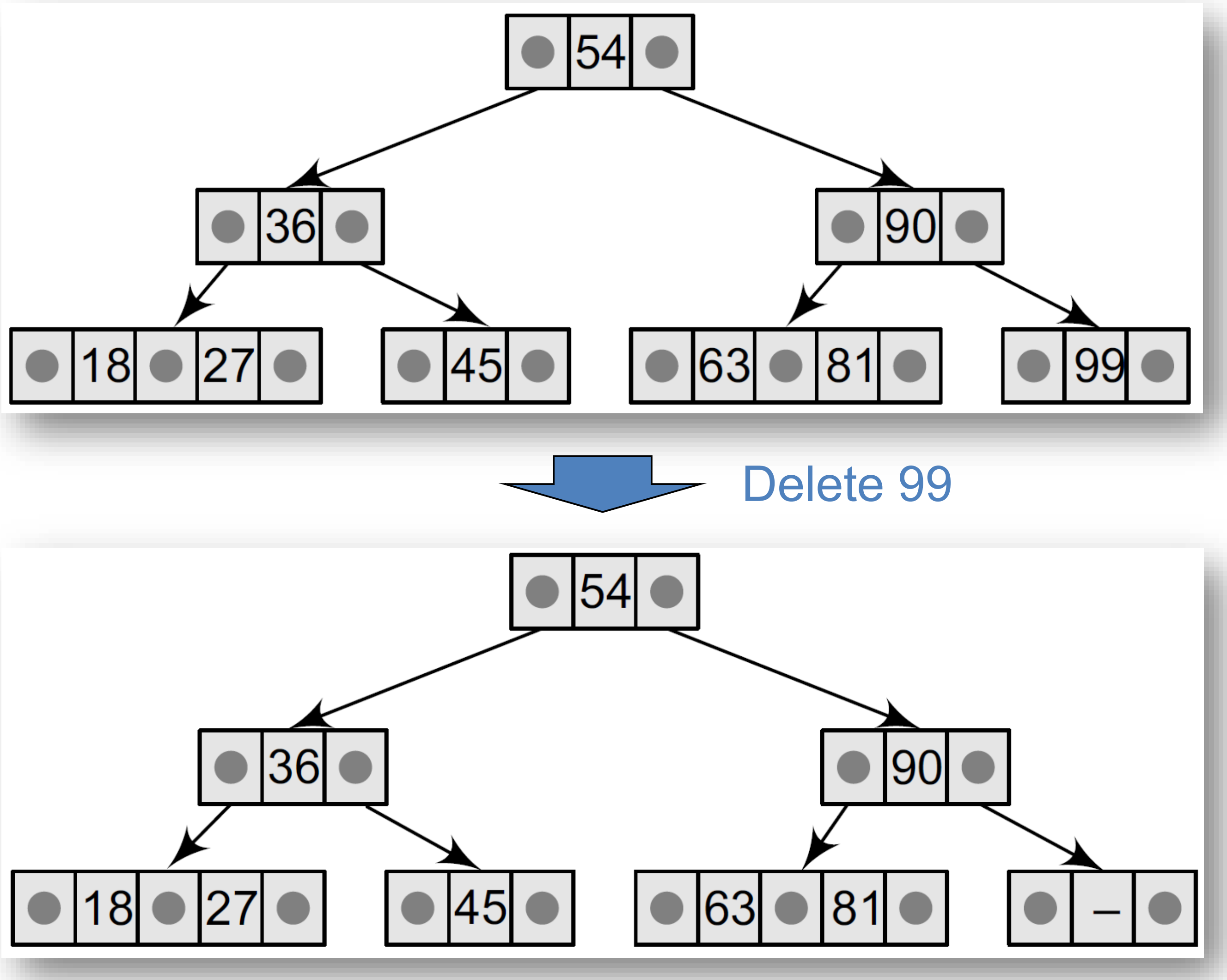


To merge the node, pull down the lowest data value in the parent's node and merge it with its left sibling



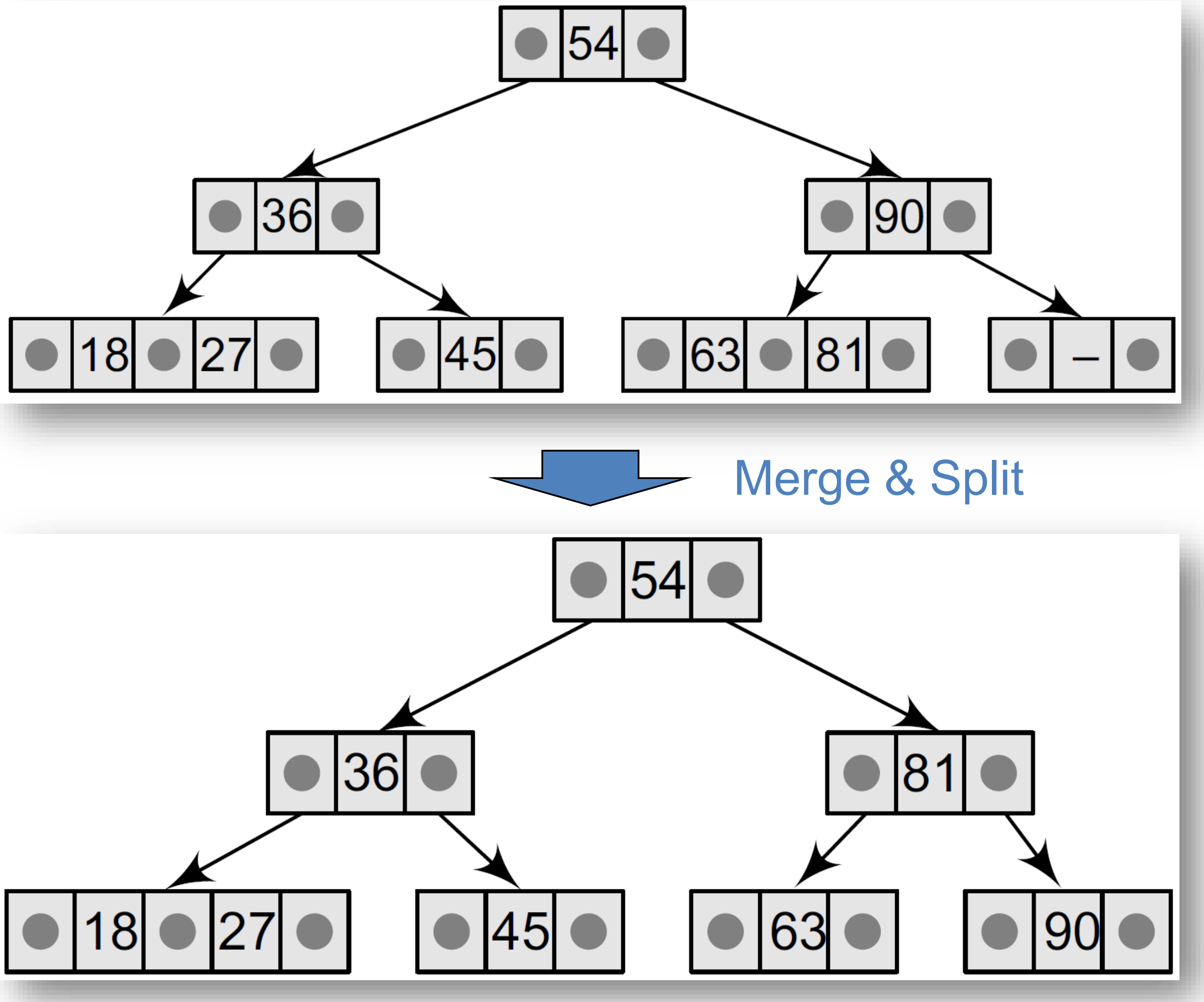
2-3 Trees – Deletion….

- Given a 2-3 tree, please delete the values 69, 72, 99, 81



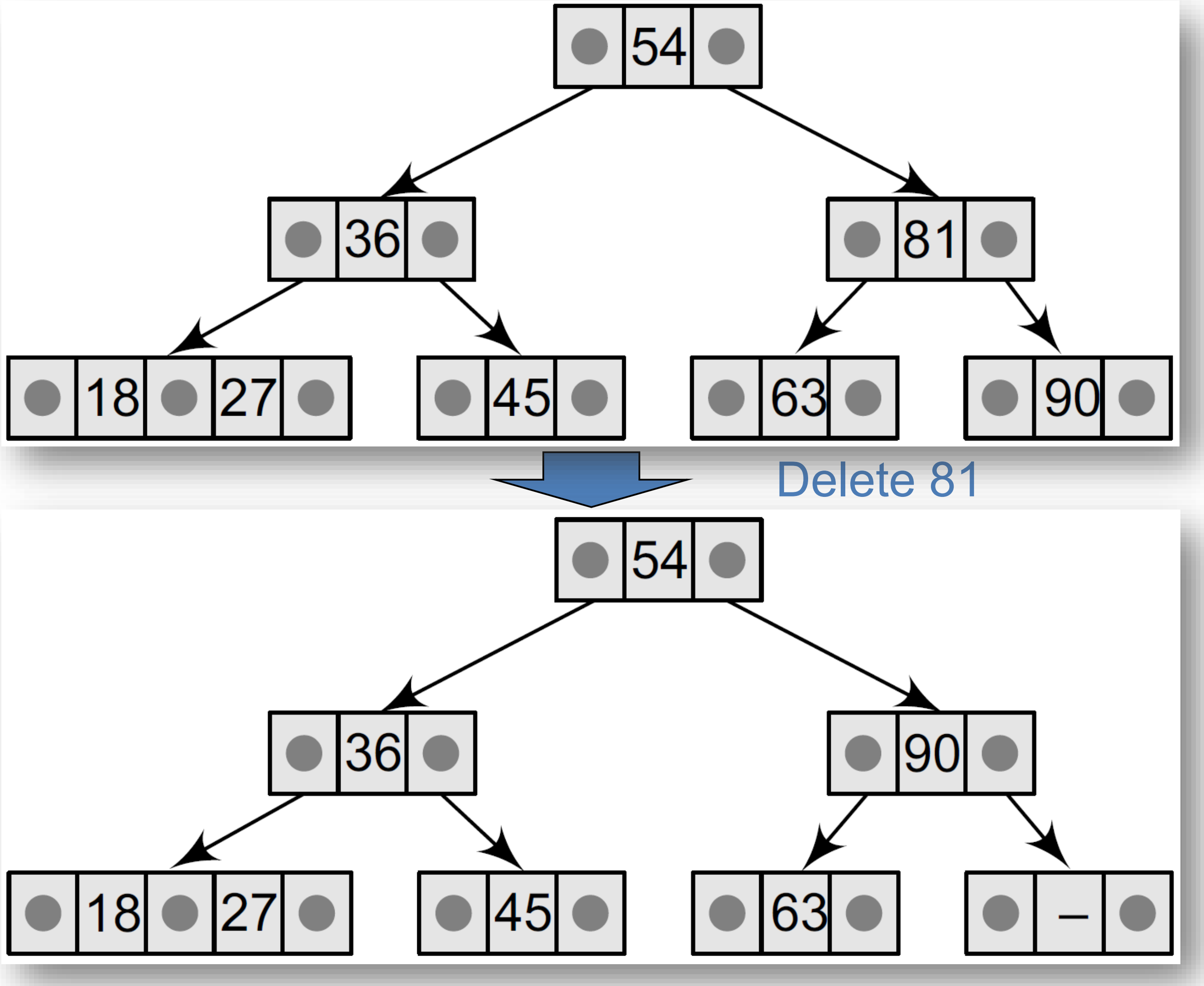
2-3 Trees – Deletion…..

- Given a 2-3 tree, please delete the values 69, 72, 99, 81



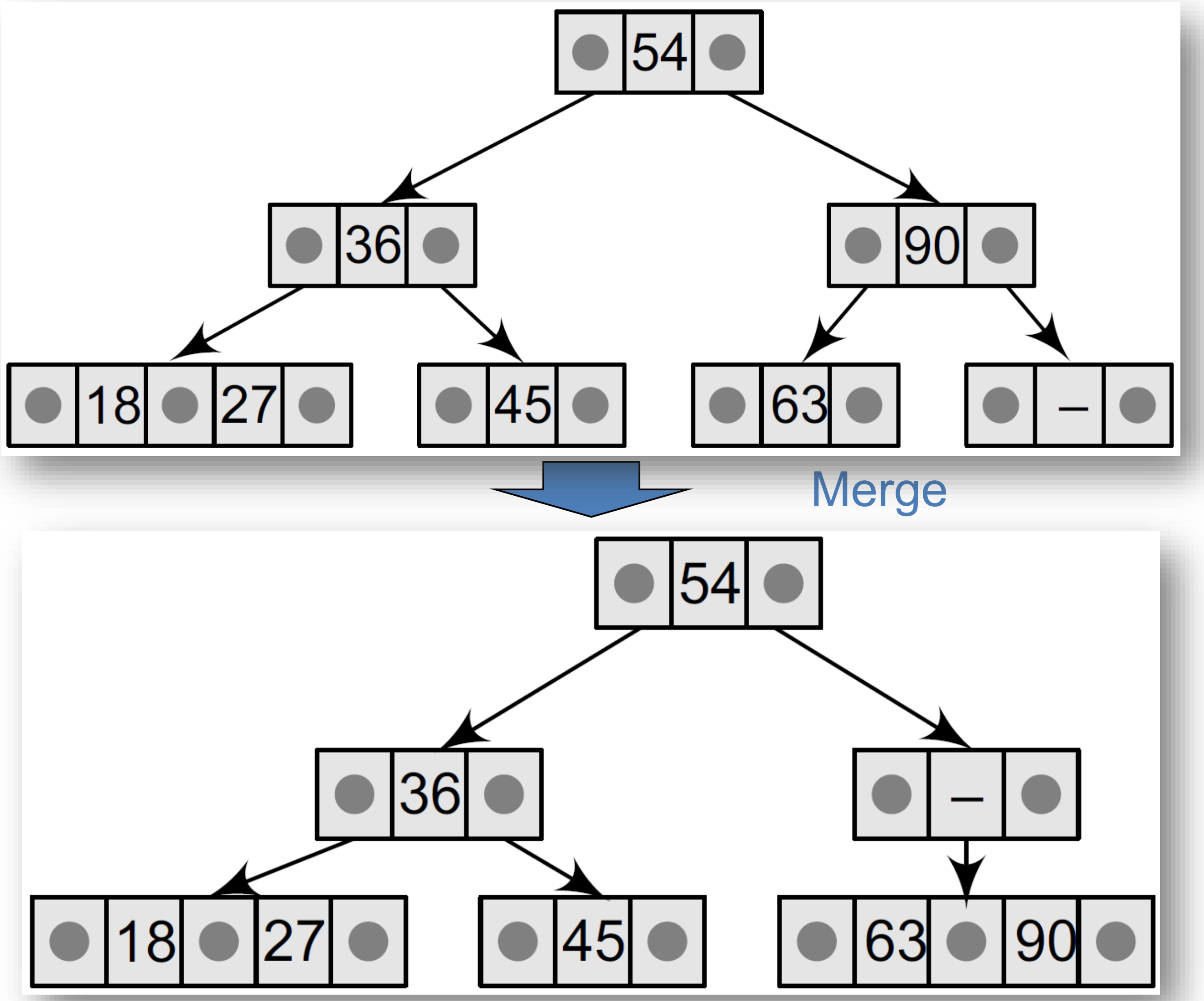
2-3 Trees – Deletion…..

- Given a 2-3 tree, please delete the values 69, 72, 99, 81



2-3 Trees – Deletion……

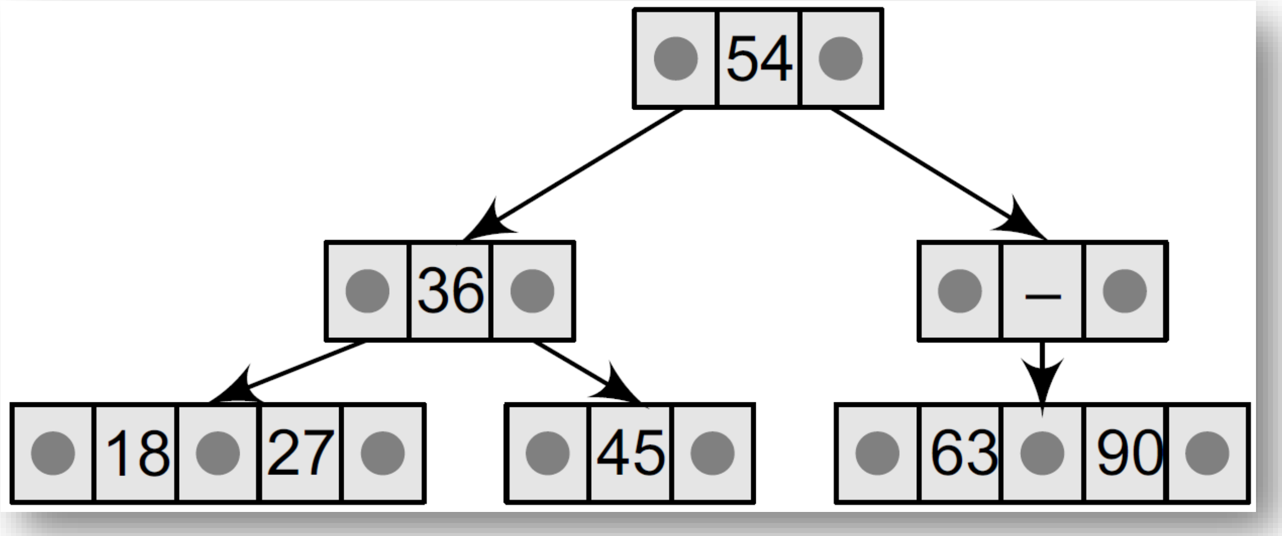
- Given a 2-3 tree, please delete the values 69, 72, 99, 81

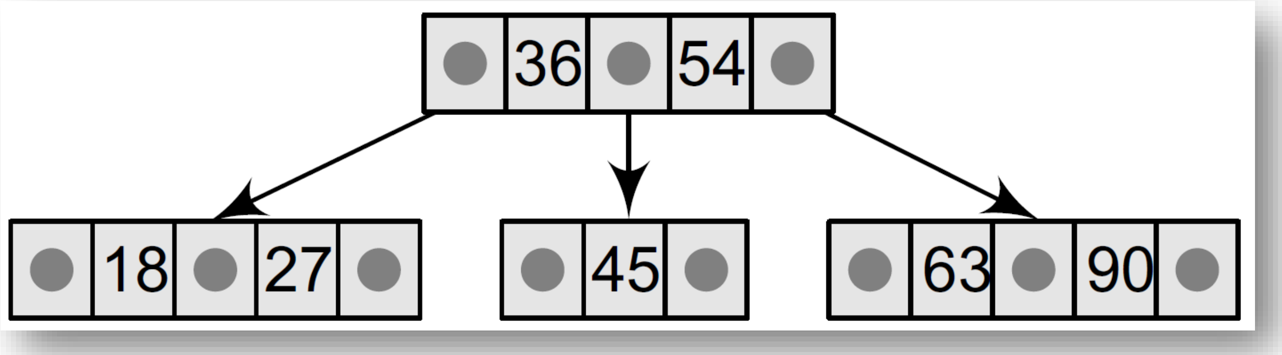


Merge

2-3 Trees – Deletion……

- Given a 2-3 tree, please delete the values 69, 72, 99, 81





Questions?

kychen@mail.ntust.edu.tw