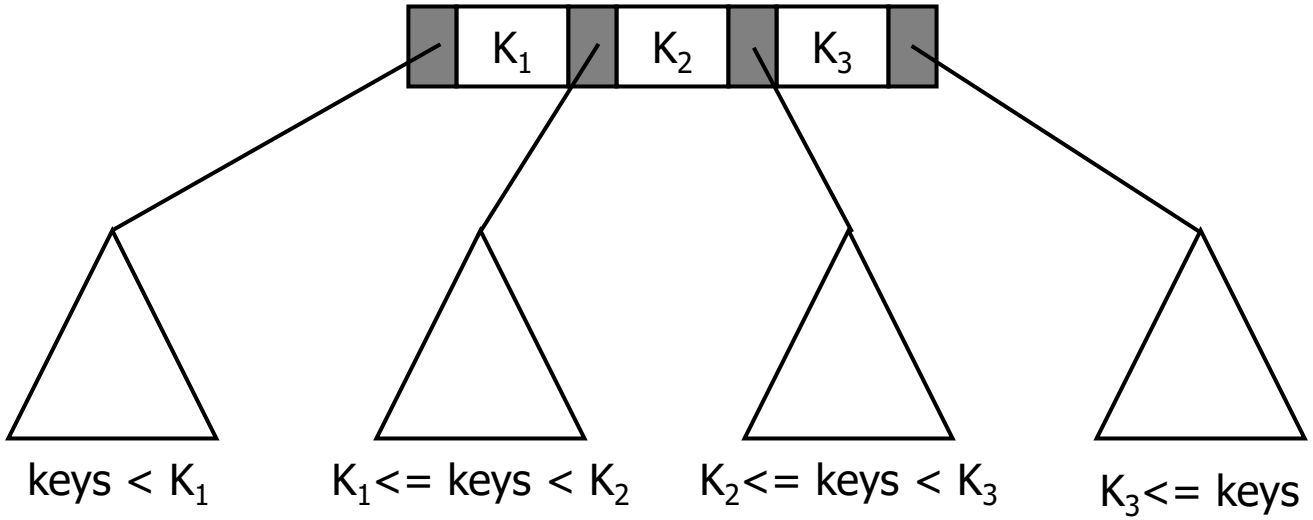
Multiway Trees

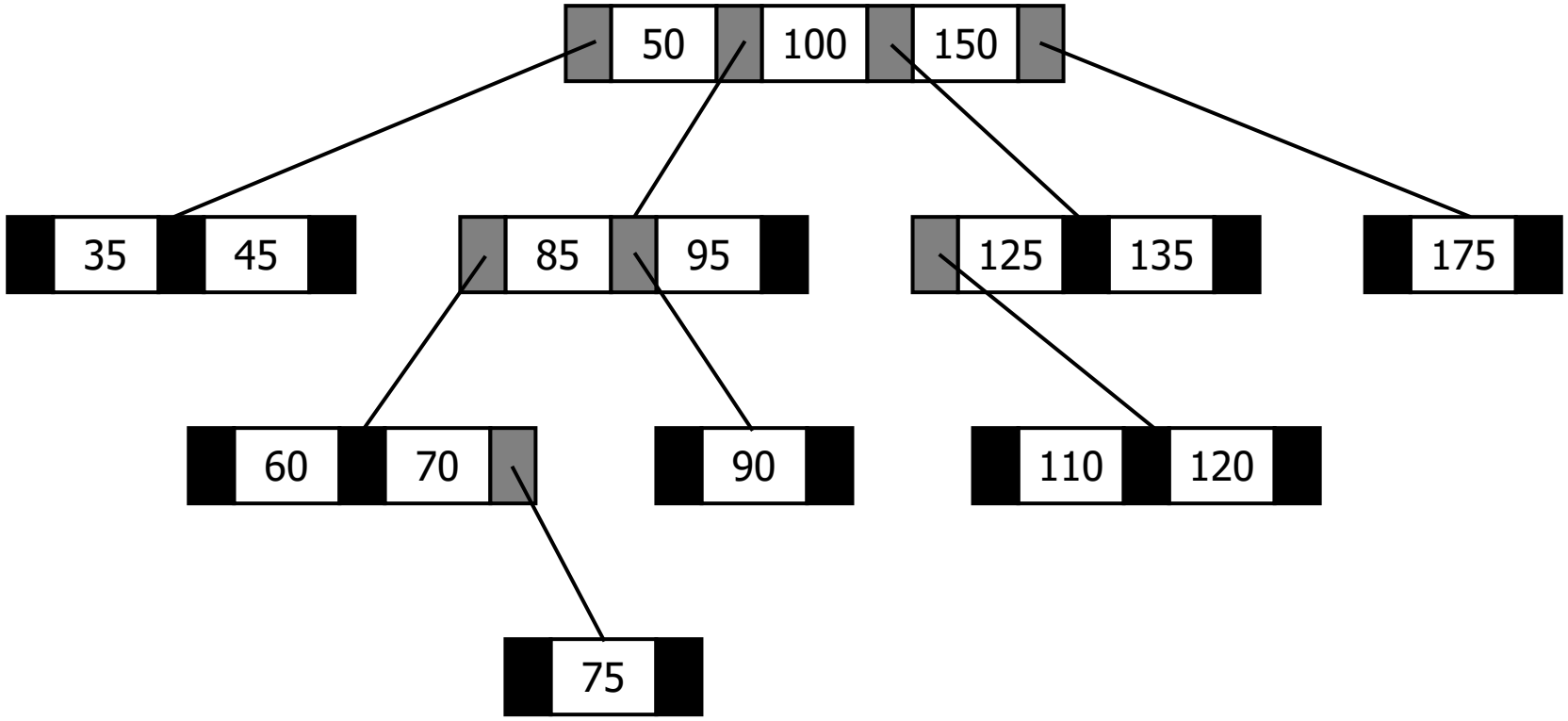
- Tree whose outdegree is not restricted to 2 while retaining the general properties of binary search trees.

M-Way Search Trees

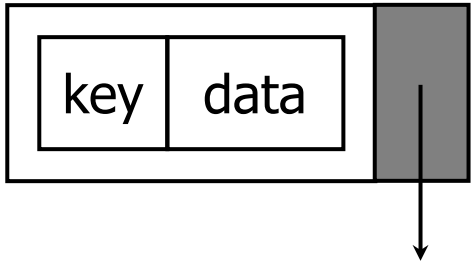
- Each node has m - 1 data entries and m subtree pointers.
- The key values in a subtree
  - >= the key of the left data entry
  - < the key of the right data entry.

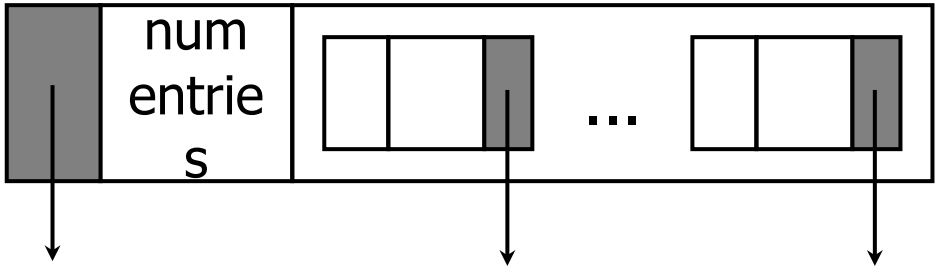


M-Way Search Trees



M-Way Node Structure






```
entry
        key <key type>
        data <data type>
        rightPtr <pointer>
end entry

node
        firstPtr <pointer>
        numEntries <integer>
        entries <array[1 .. m-1] of
entry>
end node
```

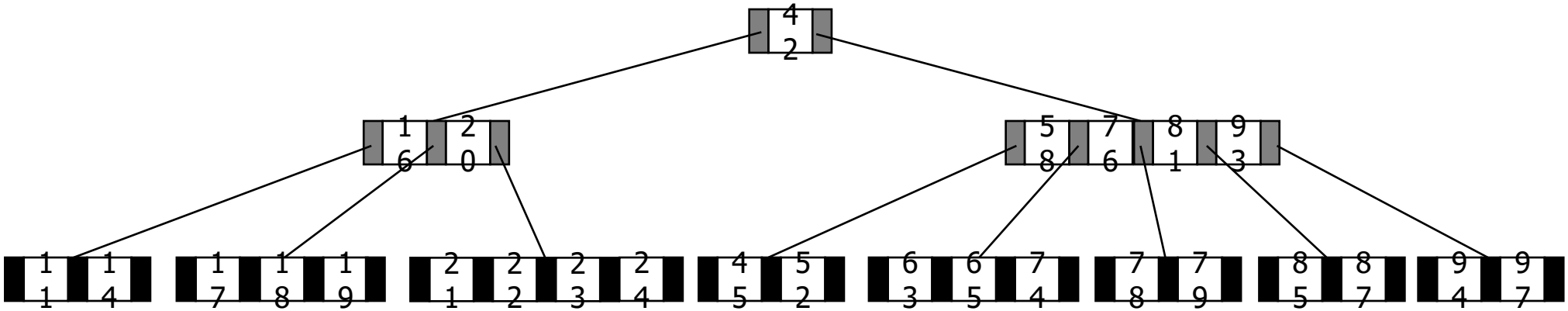
B-Trees

- M-way trees are unbalanced.
- Bayer, R. & McCreight, E. (1970) created B-Trees.

B-Trees

- A B-tree is an m-way tree with the following additional properties:
  - The root is either a leaf or has at least 2 and at most m subtrees.
  - Each node, except for the root and the leaves, has between $\lceil m/2 \rceil$ and m children
  - A leaf node has at least $\lceil m/2 \rceil - 1$ and at most m - 1 entries.
  - All leaf nodes are at the same level.

B-Trees


42
16
20
58
76
81
93
11
14
17
18
19
21
22
23
24
45
52
63
65
74
78
79
85
87
94
97

m = 5

B-Tree Insertion

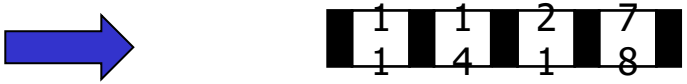
- Insert the new entry into a leaf node.
- If the leaf node is overflow, then split it and insert its median entry into its parent.
- If the internal node is overflow, do the same thing
- If the root is overflow, then create a new root containing the median entry.

B-Tree Insertion

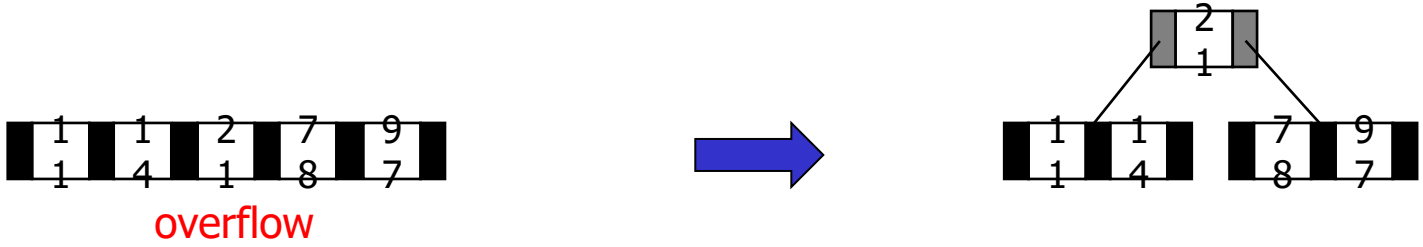
Insert 78

Insert 21

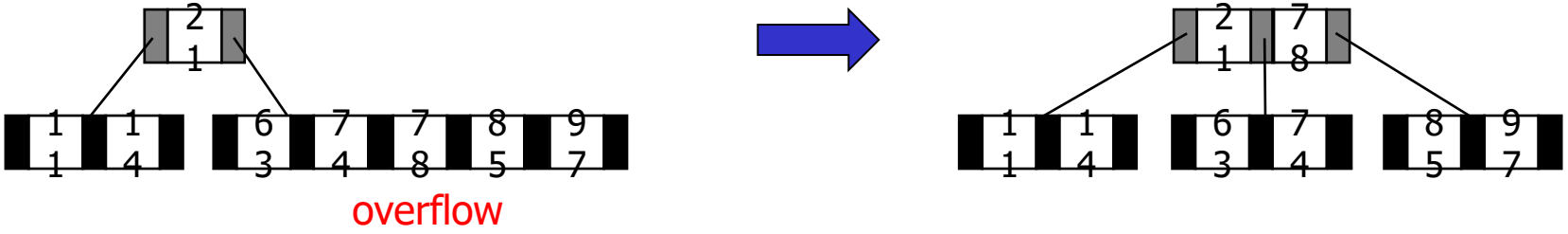
Insert 14, 11



Insert 97

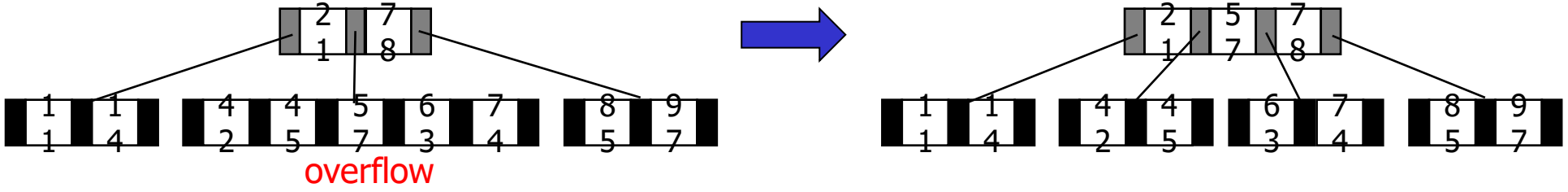


Insert 85, 74, 63

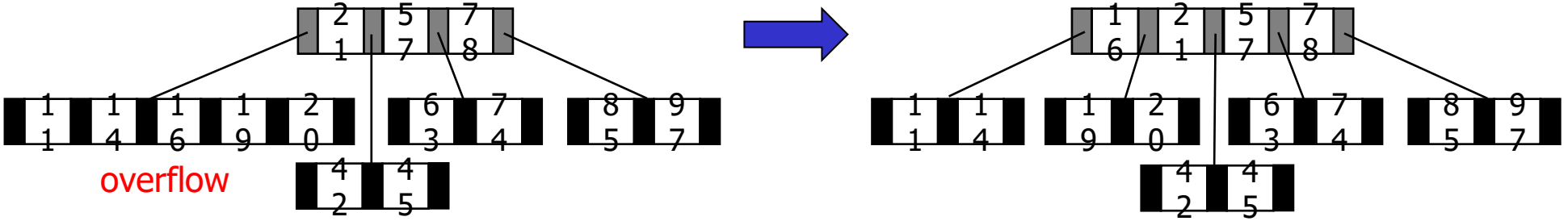


B-Tree Insertion

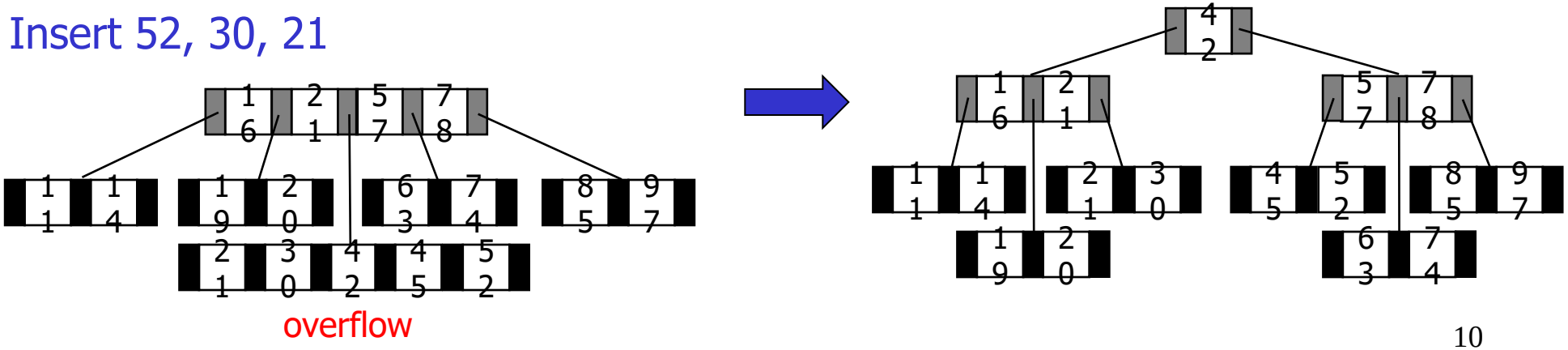
Insert 45, 42, 57



Insert 20, 16, 19



Insert 52, 30, 21

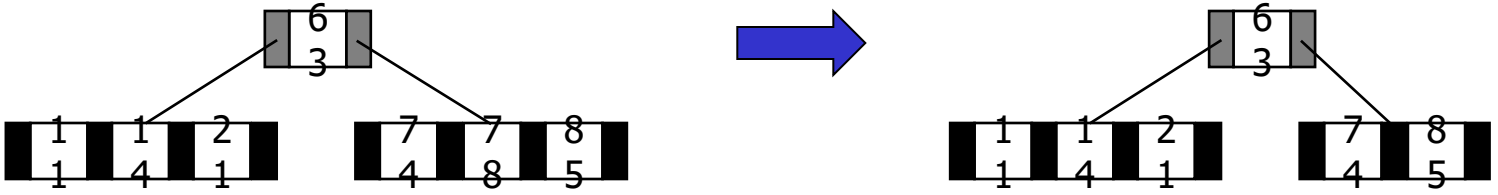


B-Tree Deletion

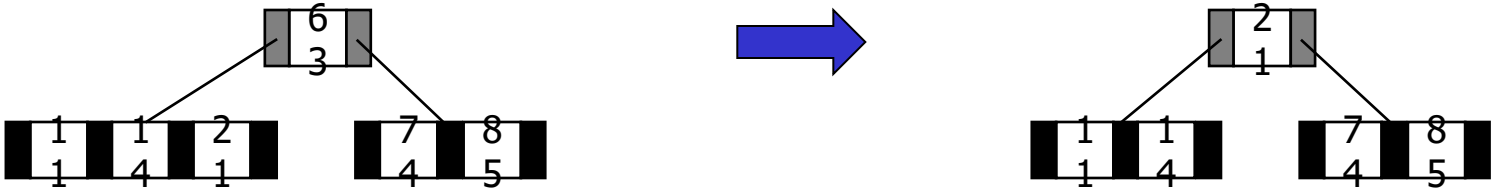
- It must take place at a leaf node.
- If the data to be deleted are not in a leaf node, then replace that entry by the largest entry on its left subtree.

B-Tree Deletion

Delete 78

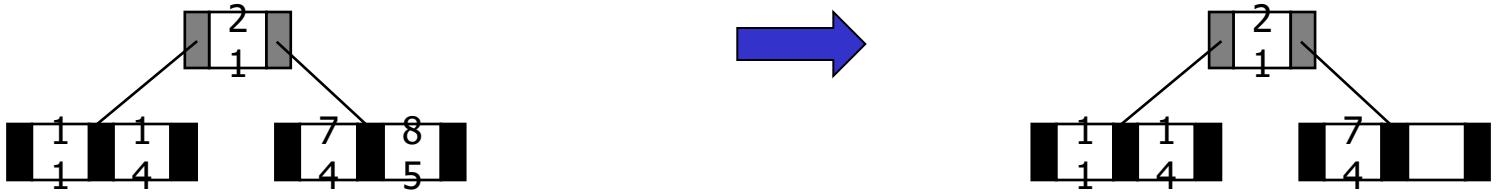


Delete 63



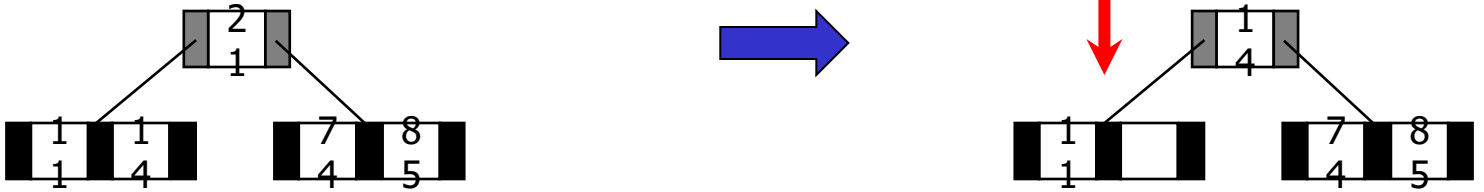
B-Tree Deletion

Delete 85



underflow
(node has fewer than the min num of entries)

Delete 21



Reflow

- For each node to have sufficient number of entries:
  - Balance: shift data among nodes.
  - Combine: join data from nodes.

Balance

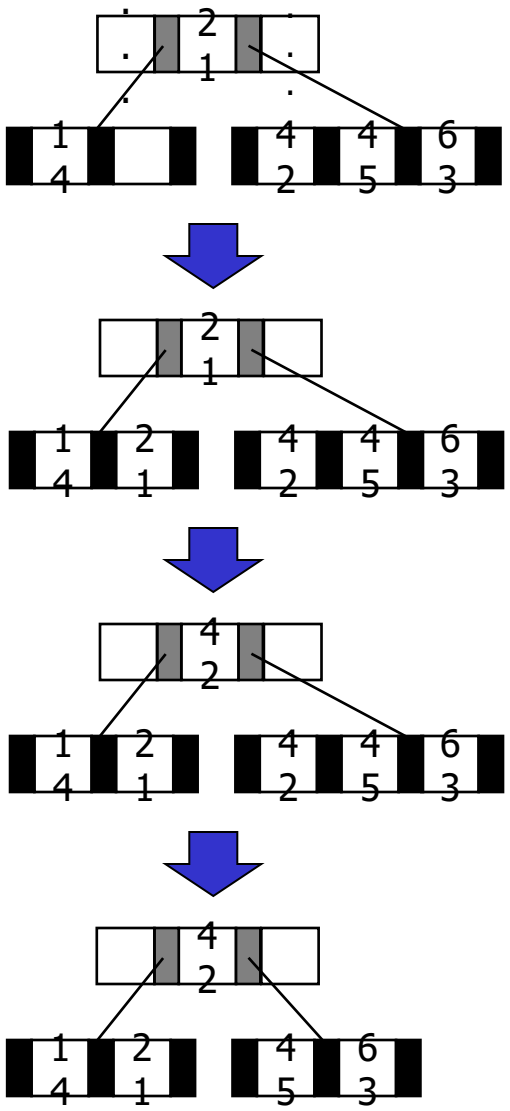
Borrow from right

Original node

Rotate parent data down

Rotate data to parent

Shift entries left



when the right sibling of the underflow node has more than min num of entries

Balance

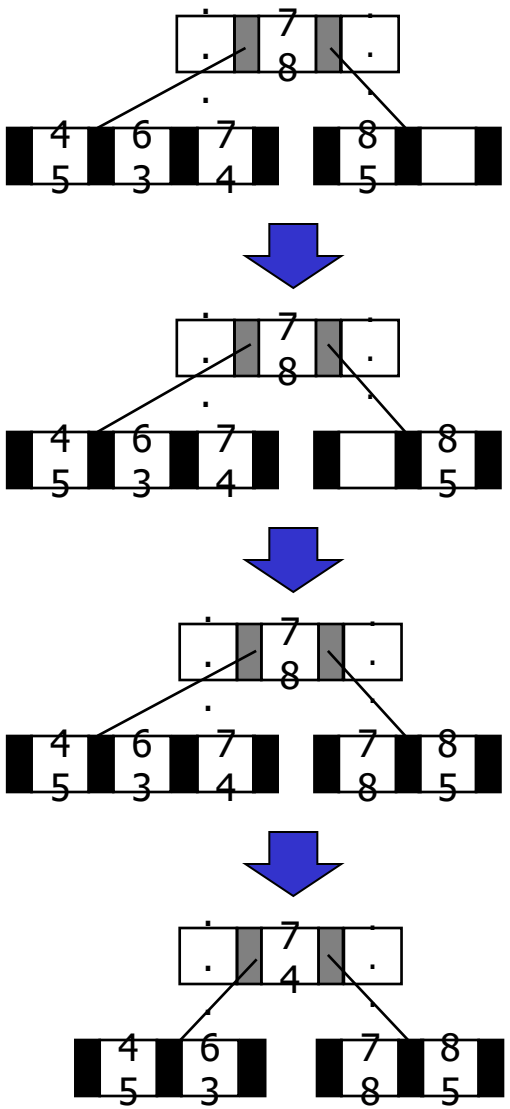
Borrow from left

Original node

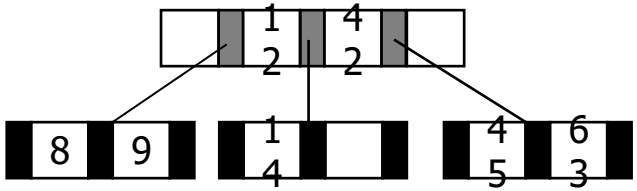
Shift entries right

Rotate parent data down

Rotate data up

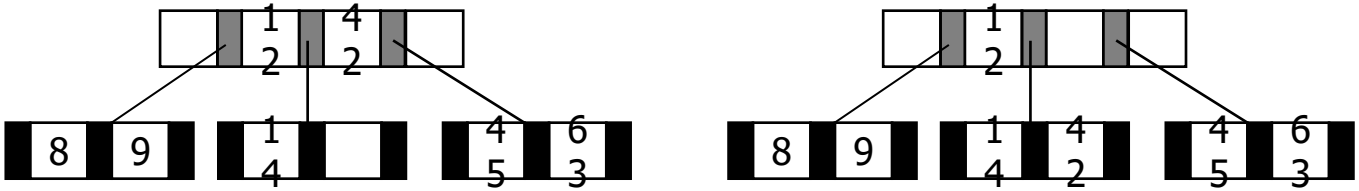


when the left sibling of the underflow node has more than min num of entries

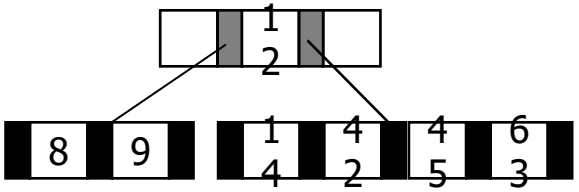


Combine

- when both left and right sibling nodes of the underflow nodes have min num of entries

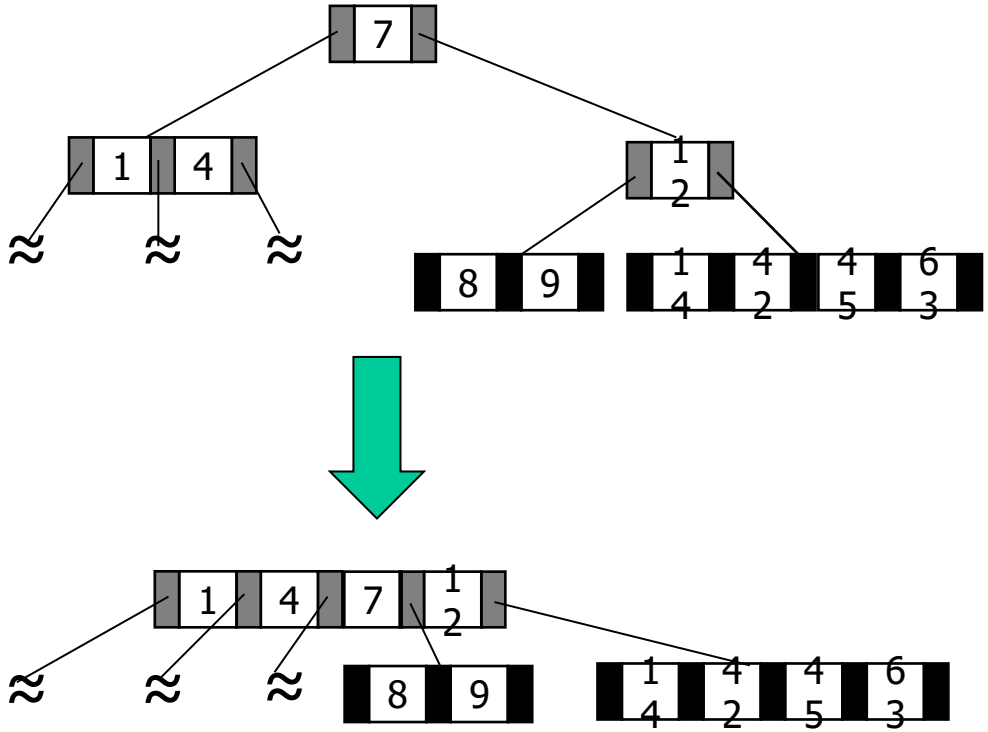


- choose one of its sibling
- move the separator down to the underflow node



- combine the underflow node with the chosen sibling
- if the parent node is underflow, repeat this combination until the root.

Combine (cont'd)



B-Tree Traversal