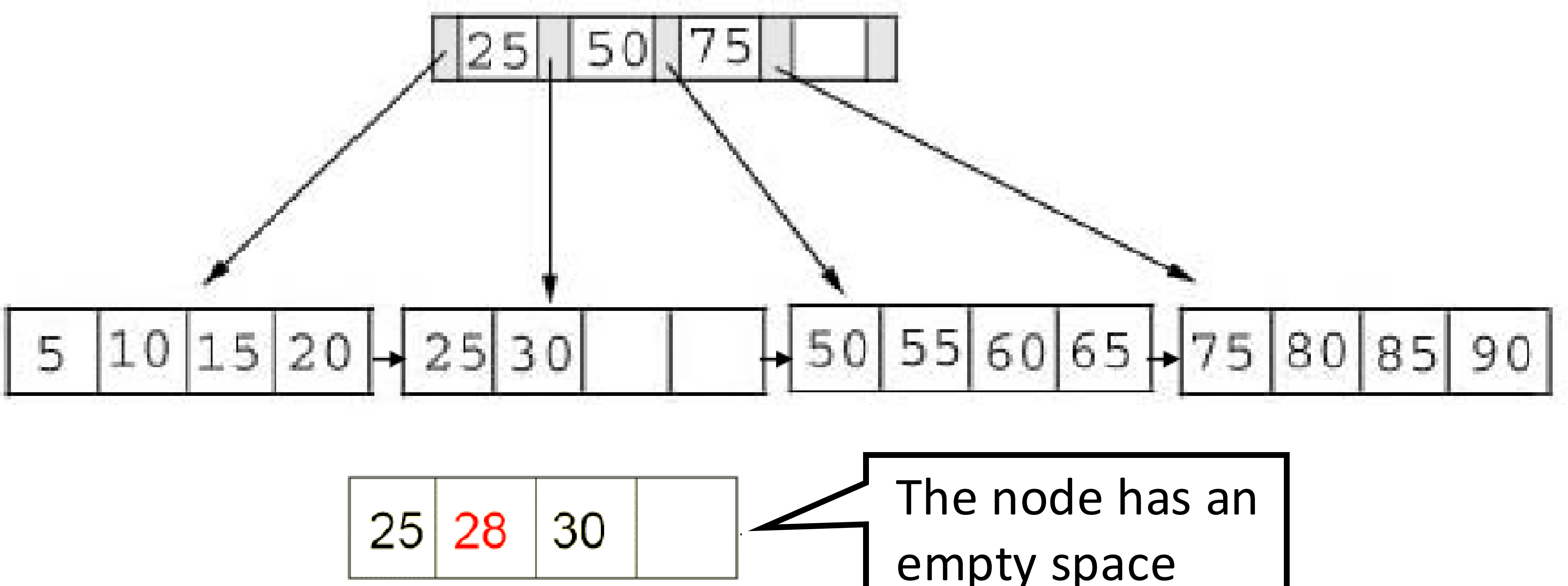
Tree data structure (8)

B+ tree index insertion

- **Step 1:** Descend to the leaf node where the key fits
- **Step 2:**
  - ❑ (Case 1): If the node has an empty space, insert the key into the node.
  - ❑ (Case 2) If the node is already full, split it into two nodes by the middle key value, distributing the keys evenly between the two nodes, so each node is half full.
    - (Case 2a) If the node is a leaf, take a copy of the middle key value, and repeat step 2 to insert it into the parent node.
    - (Case 2b) If the node is a non-leaf, exclude the middle key value during the split and repeat step 2 to insert this excluded value into the parent node.

Tree data structure (9)

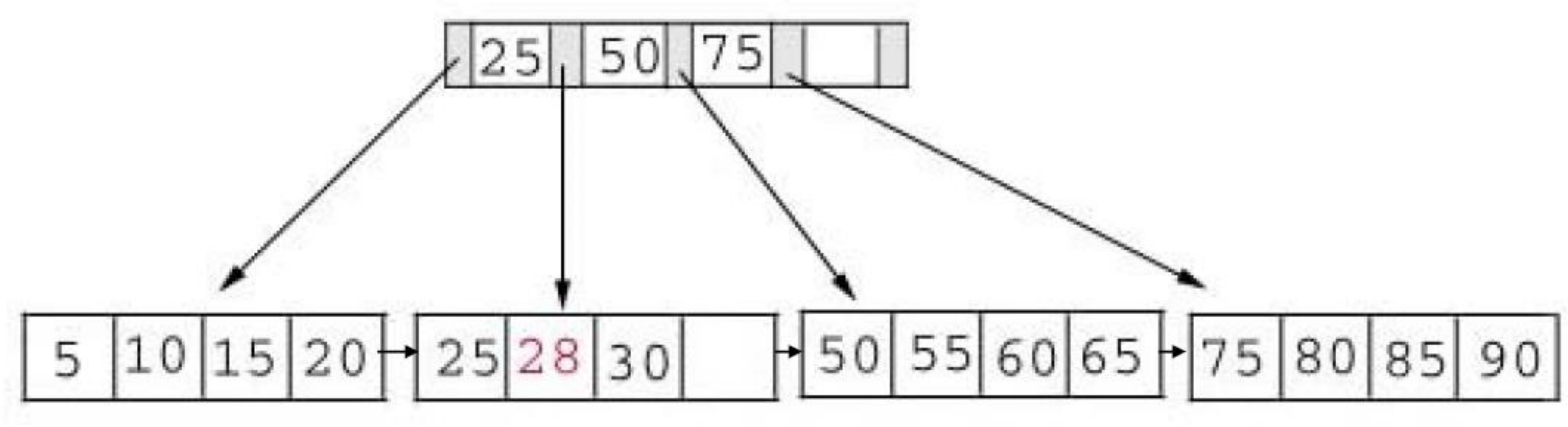
B+ tree index insertion – example 1: insert 28 into the tree below



The node has an empty space

Tree data structure (10)

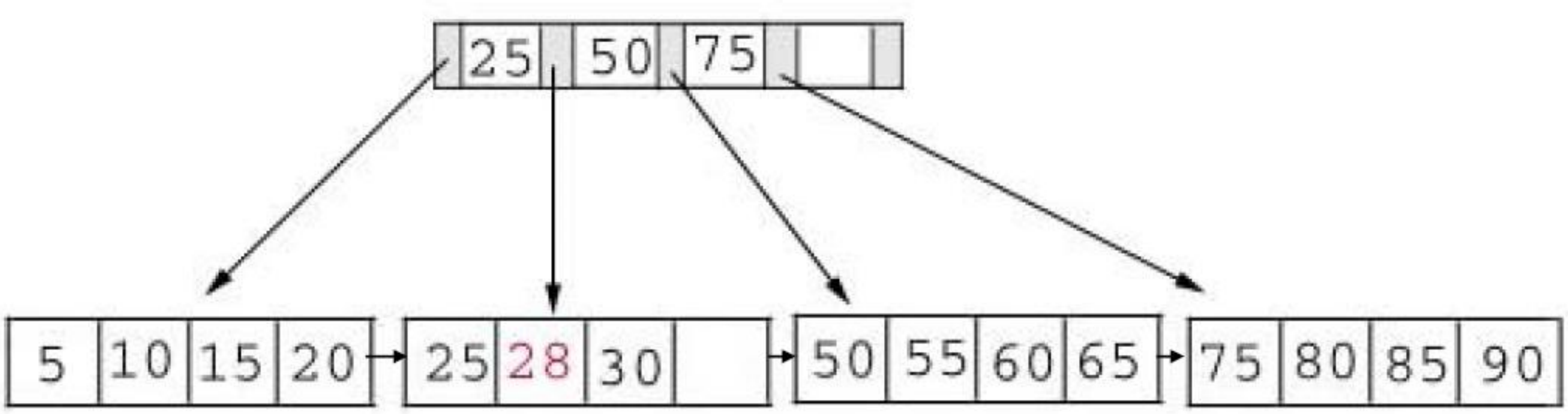
B+ tree index insertion – example 1: insert 28 into the tree below



Insert 28 into the appropriation leaf node.

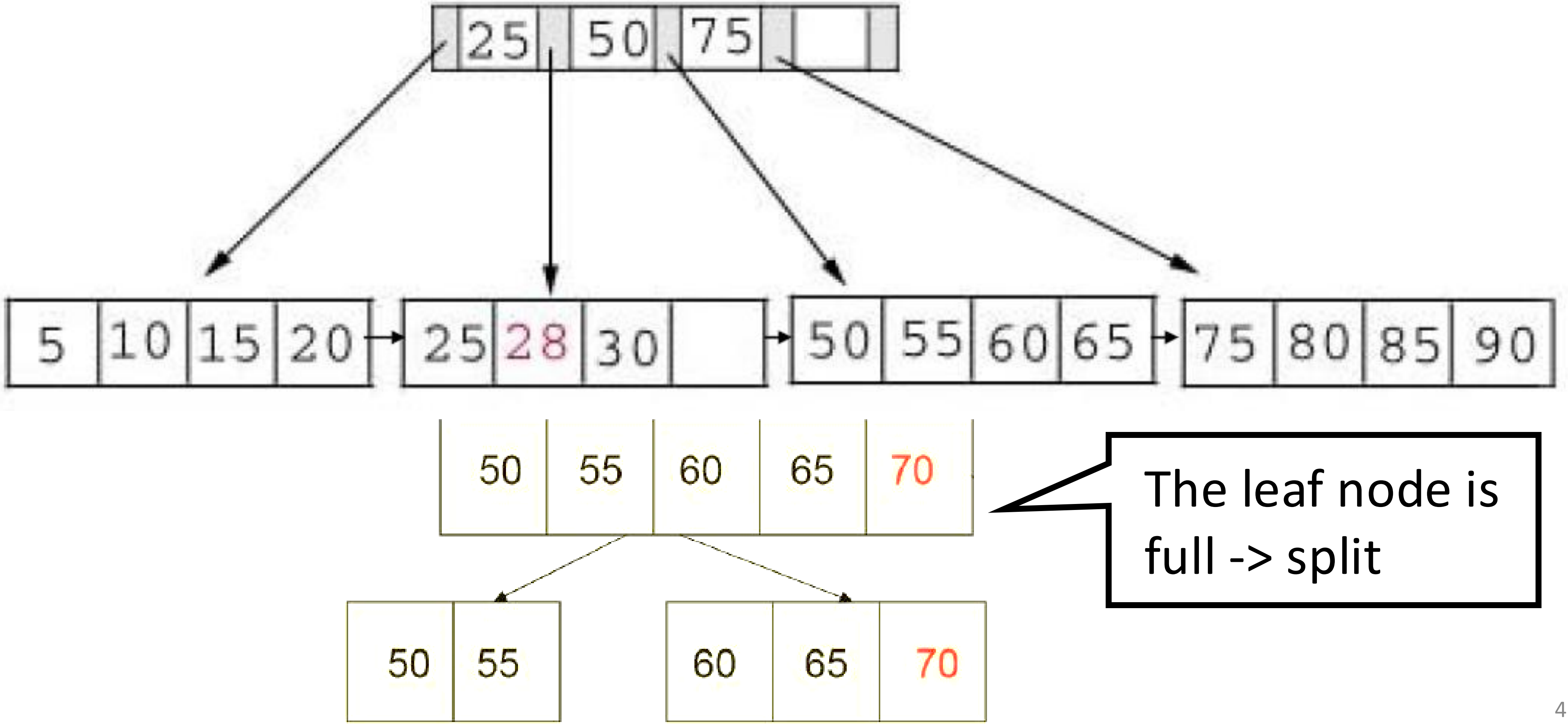
Tree data structure (11)

B+ tree index insertion – example 2: insert 70 into the tree below



Tree data structure (13)

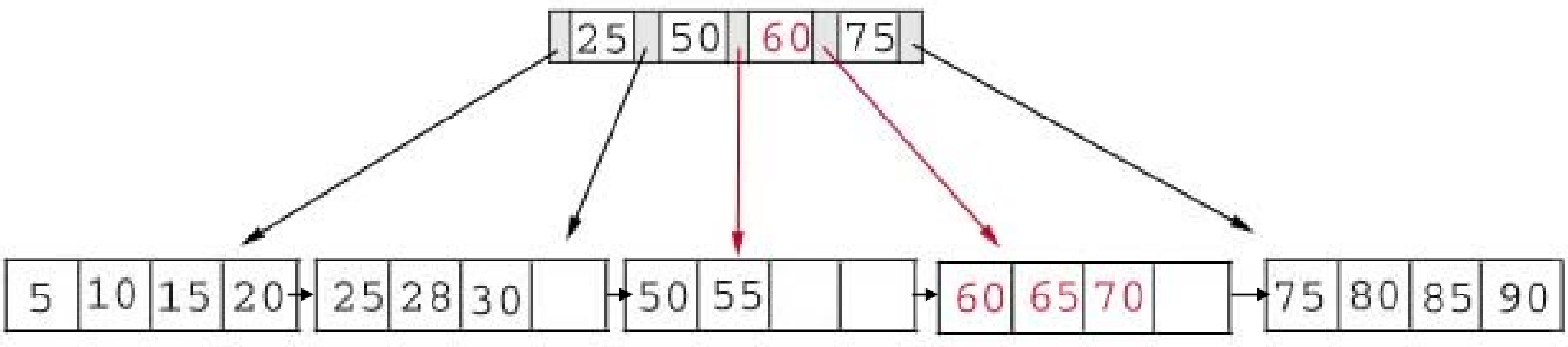
B+ tree index insertion – example 2: insert 70 into the tree below



The leaf node is full -> split

Tree data structure (14)

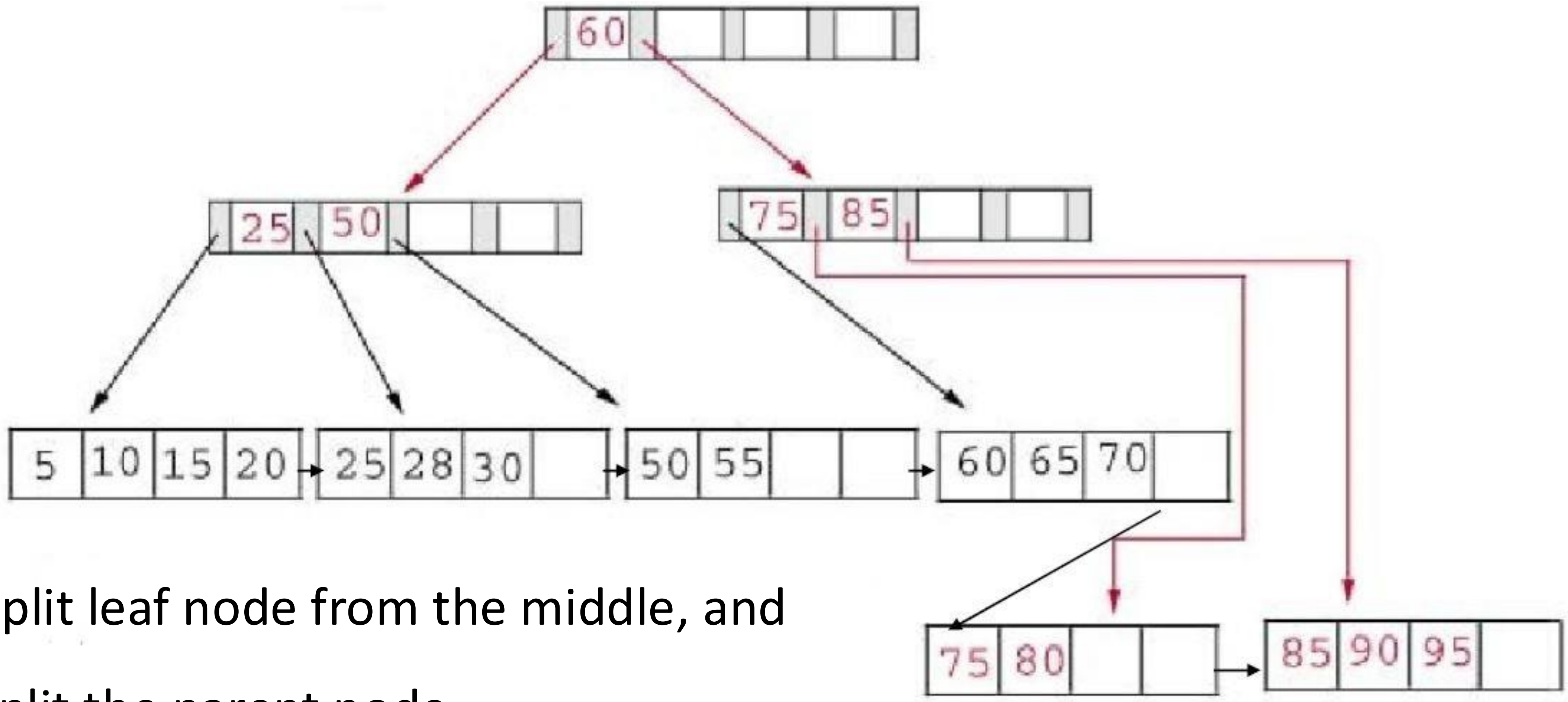
B+ tree index insertion – example 2: insert 70 into the tree below



Split leaf node from the middle, chose the middle key 60, and place it into the parent node.

Tree data structure (15)

B+ tree index insertion – example 3: insert 95 into the tree below



Split leaf node from the middle, and

split the parent node.