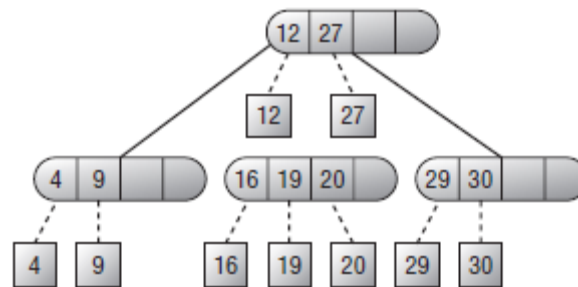


# B-Trees

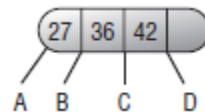


# Agenda

- B-Trees
- Balanced Tree Variations
- Summary
- Exercises

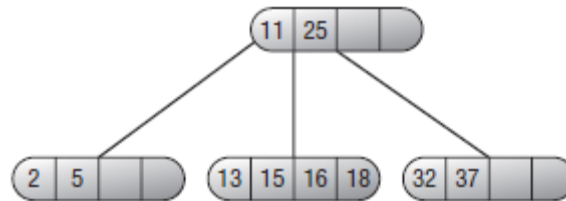
# B-Trees

- B-trees extension of 2-3 trees. (Or, if you prefer, 2-3 trees are a special case of B-trees.)
- In a B-tree of order  $K$ , every internal node (except possibly the root) holds between  $K$  and  $2 \times K$  values and has between  $K + 1$  and  $2 \times K + 1$  branches



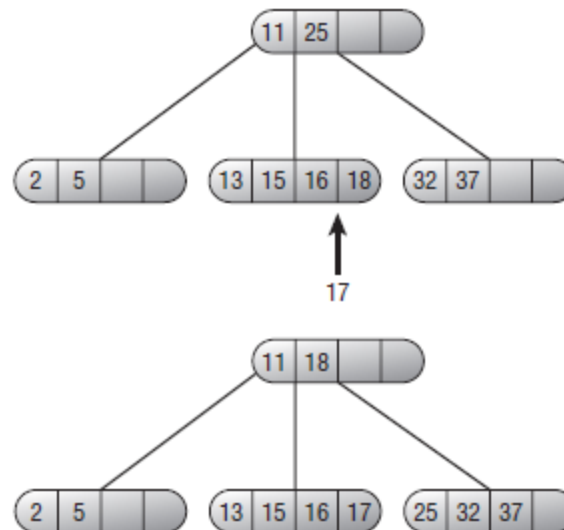
# B-Tree Properties

- A B-tree of order K has these properties:
  - Each node (except possibly the root) holds between K and  $2 \times K$  values
  - An internal node holding M values has  $M + 1$  branches leading to  $M + 1$  children
  - All leaves are at the same level



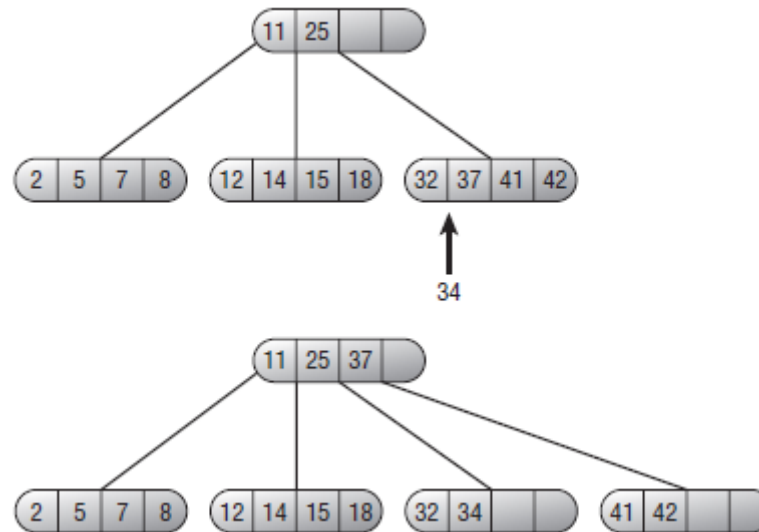
# Adding Values

- Sometimes you can redistribute values among siblings



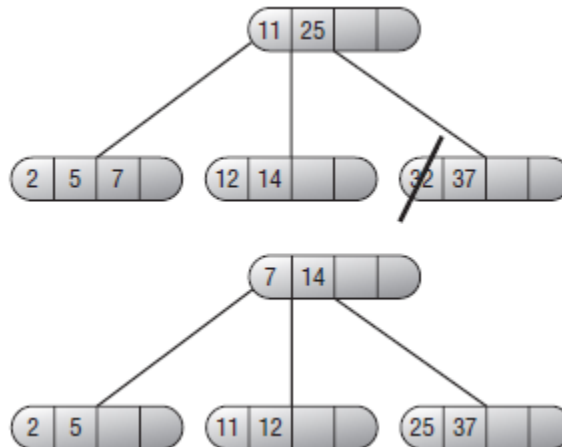
# Adding Values (continued)

- If siblings are full, you must split a bucket



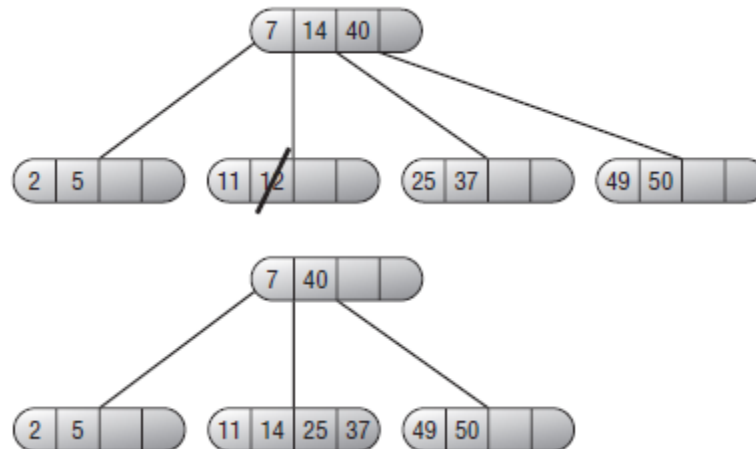
# Deleting Values

- Sometimes you can redistribute values among siblings



# Deleting Values (continued)

- If siblings are too empty, you must merge buckets





# Balanced Tree Variations

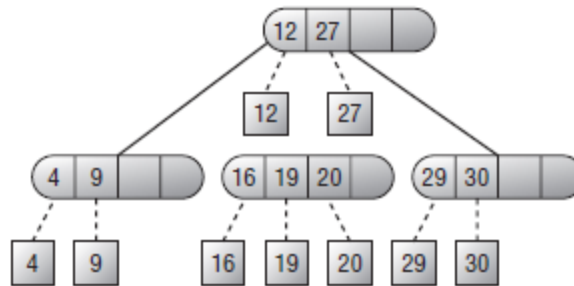
## Top-Down B-trees

- Split full nodes on the way down the tree

# Balanced Tree Variations

## B+trees

- Store data outside of the nodes



# Summary

- B-Trees
  - Adding values
  - Deleting values
- Balanced Tree Variations
  - Top-Down B-trees
  - B+trees

# Exercises

- Chapter 11 Exercises 6 – 9.
- Read *Essential Algorithms, 2e* Chapter 12 pages 367 – 381. (Stop before the section “Decision Tree Heuristics.”)