

Multi-way Search Trees

COMP 2210 – Dr. Hendrix



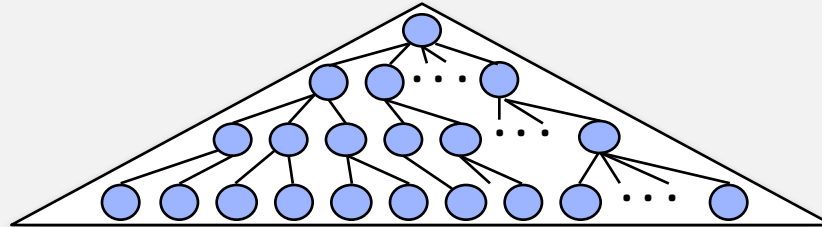
AUBURN

UNIVERSITY

SAMUEL GINN
COLLEGE OF ENGINEERING

Multi-way search trees

A multi-way search tree (an **M-way tree**) is a tree of order $M > 2$ in which the search property (total order) holds on every node and in which all leaves are at the same depth.

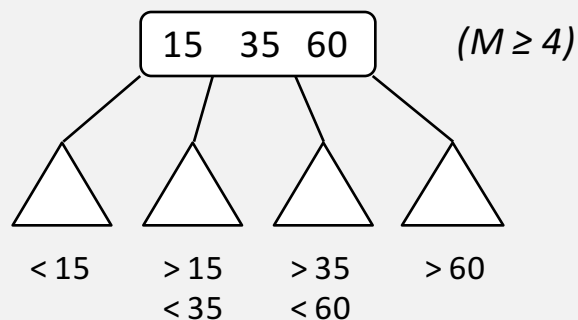


In an M-way tree:

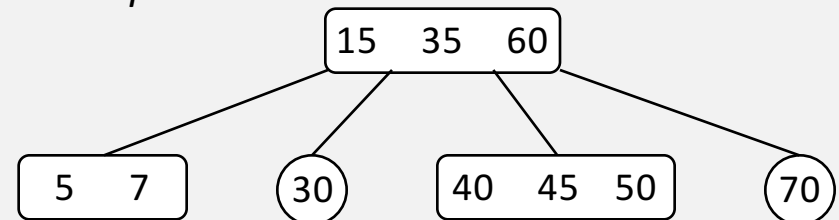
Each node holds between 1 and $M-1$ values in sorted order.

A non-leaf node with K values has $K+1$ non-empty subtrees that are M-way search trees.

The i -th subtree of a node that holds values $[v_0..v_k]$ ($0 \leq i \leq K$) can only store values v such that $v_{i-1} < v < v_i$.



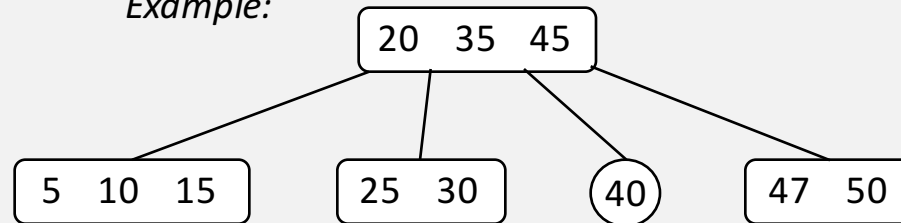
Example:



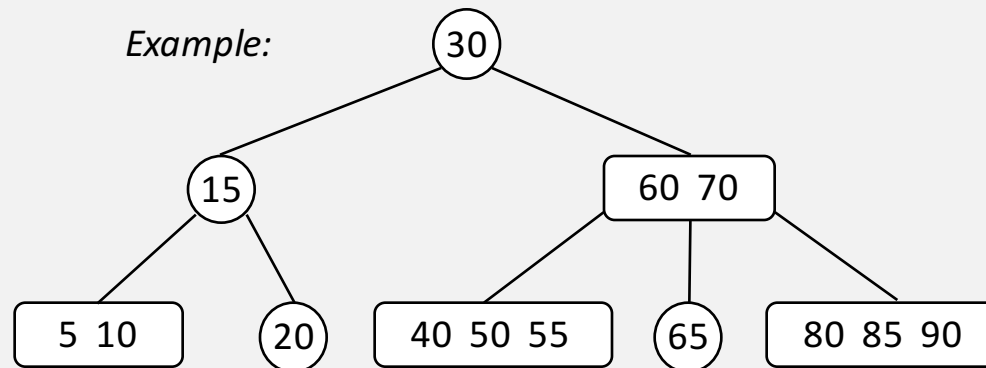
2-4 Trees

A 2-4 tree is a 4-way search tree where each non-leaf node must have at least two non-empty subtrees.

Example:



Example:



2-4 Trees – adding values

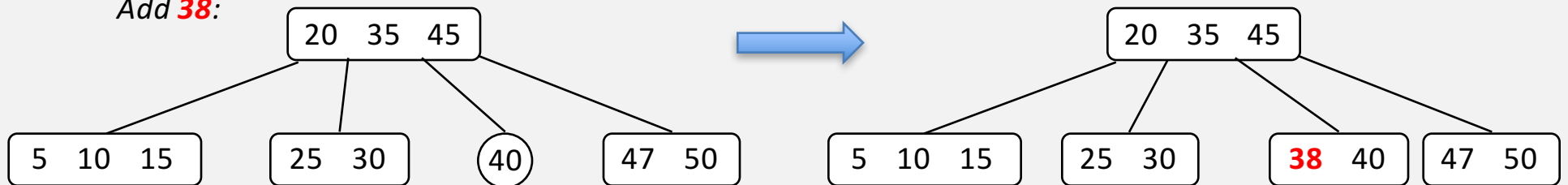
To add a new value, use the total order to find the **leaf** that should hold this value.

New values are always added in the context of an existing leaf node.

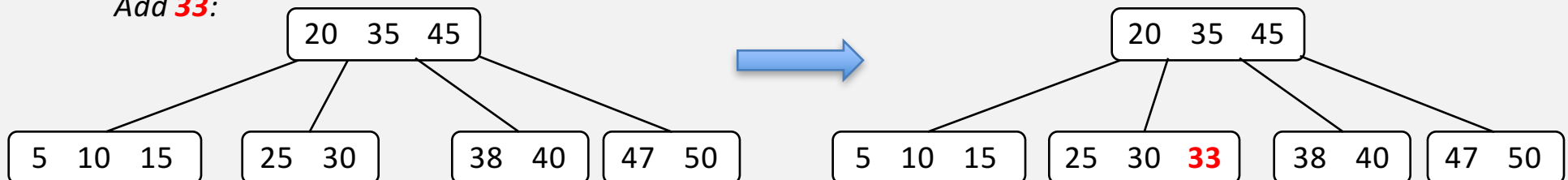


*New values in a BST went into a **new leaf** in a currently empty subtree.*

Add **38**:

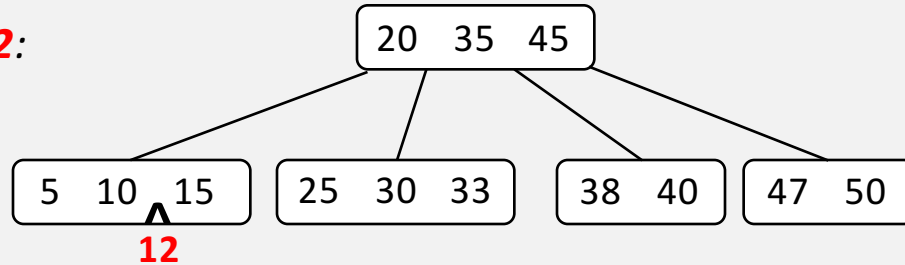


Add **33**:



2-4 Trees – growing taller

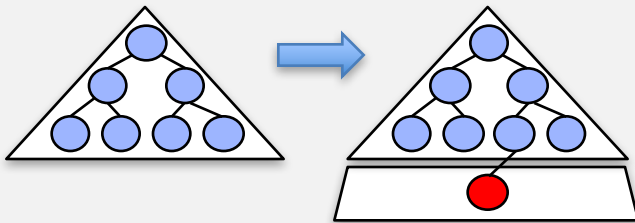
Add **12**:



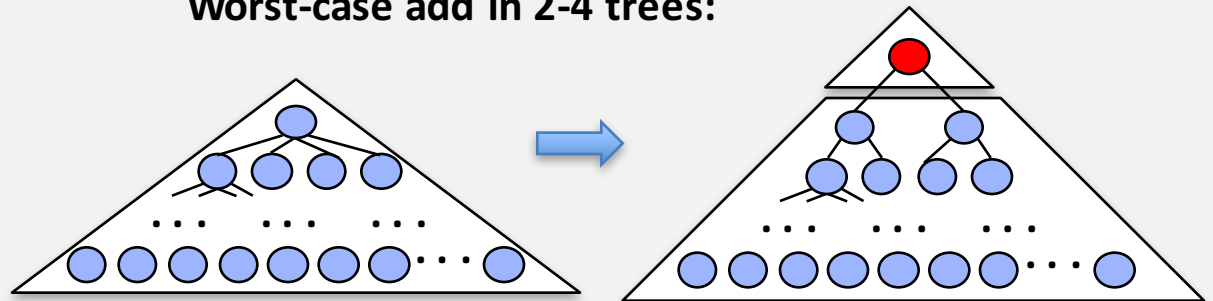
M-way search trees grow taller in a different way than BSTs.

A node in a 2-4 tree can store up to 3 values, so this leaf is **full**.

Worst-case add in balanced BSTs:



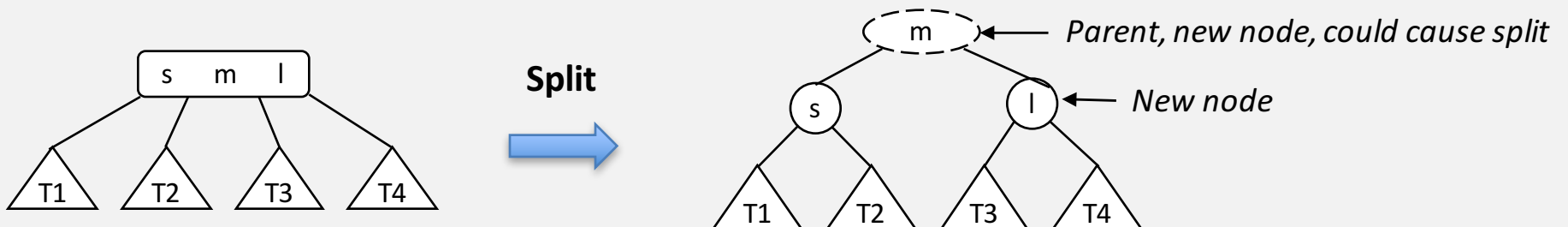
Worst-case add in 2-4 trees:



All leaves in a 2-4 tree have to be at the same level.

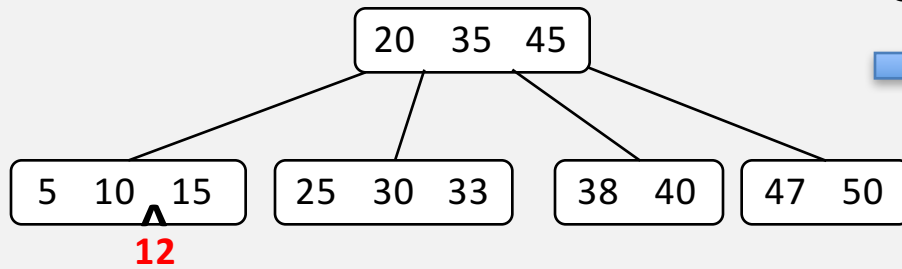
The tree grows “up” by adding a new root rather than down by adding a new (lower) leaf.

When a 2-4 node is full but it needs to store another value, we perform a **split**.

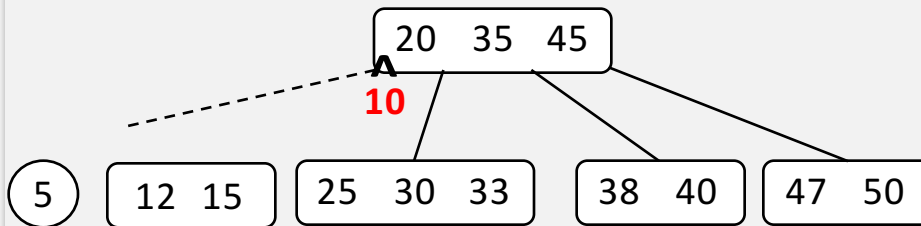
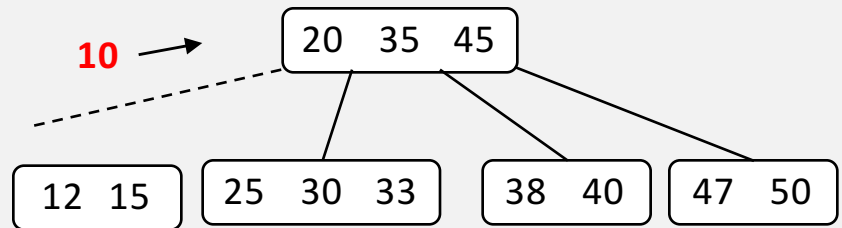
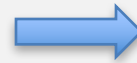


2-4 Trees – growing taller

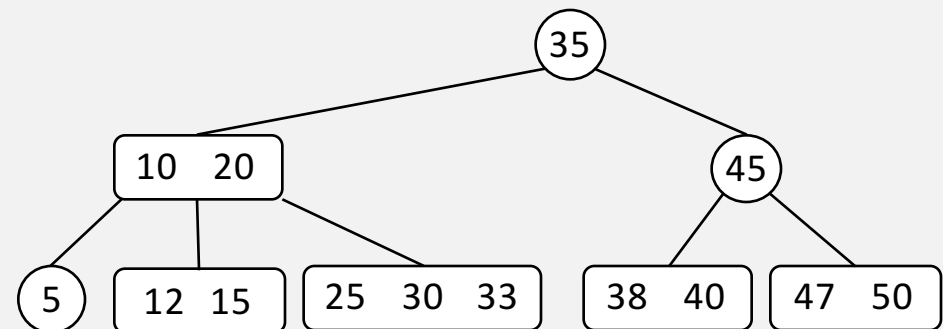
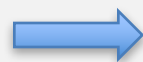
Add **12**:



Split



Split



Growing a 2-4 Tree

Insert: 10, 85, 15, 70, 20, 60, 30, 50, 65, 80, 90, 40, 5, 55

