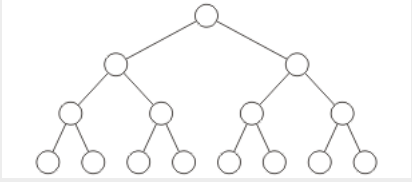
DATA STRUCTURES AND ALGORITHMS

LECTURE 16
BALANCED TREES

IMRAN IHSAN
ASSISTANT PROFESSOR
AIR UNIVERSITY, ISLAMABAD

PERFECT TREE
EXAMPLE

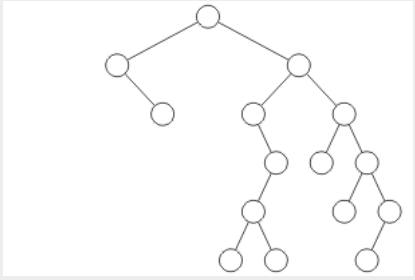
- For a perfect tree, all nodes have the same number of descendants on each side



- Perfect binary trees are balanced while linked lists are not

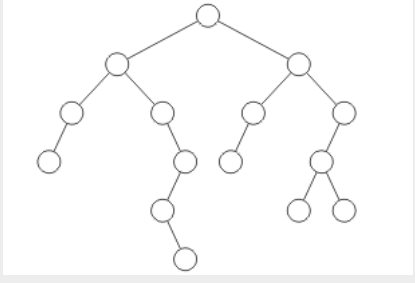
BINARY TREE
EXAMPLE

- This binary tree would also probably not be considered to be “balanced” at the root node



BINARY TREE
EXAMPLE

- How about this example?
  - The root seems balanced, but what about the left sub-tree?

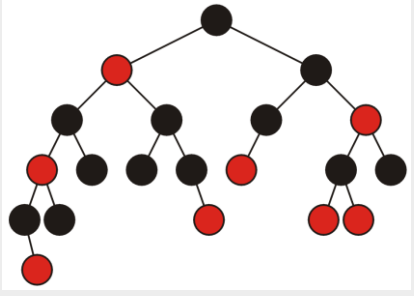


DEFINITION FOR BALANCE

- We must develop a quantitative definition of balance which can be applied
- Balanced may be defined by:
  - Height balancing: comparing the heights of the two sub trees
  - Null-path-length balancing: comparing the null-path-length of each of the two sub-trees (the length to the closest null sub-tree/empty node)
  - Weight balancing: comparing the number of null sub-trees in each of the two sub trees
- We will have to mathematically prove that if a tree satisfies the definition of balance, its height is Θ (ln(n))
- We will see one definition of height balancing:
  - AVL trees
- We will also look at B+-trees
  - Balanced trees, but not binary trees

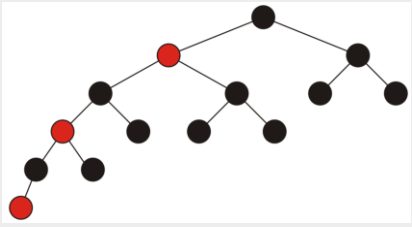
RED-BLACK TREES

- Red-black trees maintain balance by
  - All nodes are colored red or black (0 or 1)
- Requirements:
  - The root must be black
  - All children of a red node must be black
  - Any path from the root to an empty node must have the same number of black nodes



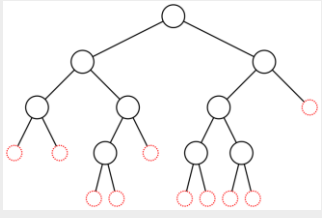
RED-BLACK TREES

- Red-black trees are null-path-length balanced in that the null-path length going through one sub-tree must not be greater than twice the null-path length going through the other
- A perfect tree of height h has a null-path length of h + 1
- Any other tree of height h must have a null-path-length less than h + 1



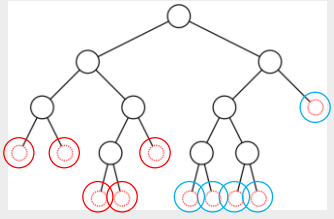
WEIGHT-BALANCED TREES

- Recall: an empty node/null subtree is any position within a binary tree that could be filled with the next insertion:
  - This tree has 9 nodes and 10 empty nodes:



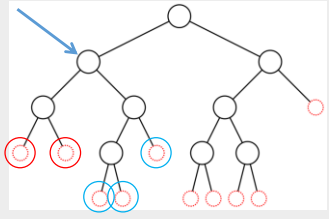
WEIGHT-BALANCED TREES

- The ratios of the empty nodes at the root node are 5/10 and 5/10



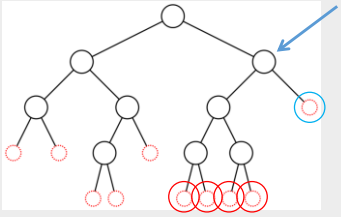
WEIGHT-BALANCED TREES

- The ratios of the empty nodes at this node are 2/5 and 3/5



WEIGHT-BALANCED TREES

- The ratios of the empty nodes at this node, however, are 4/5 and 1/5



WEIGHT-BALANCED TREES

- BB(α) trees ($0 < \alpha \leq 1/3$) maintain weight balance requiring that neither side has less than a α proportion of the empty nodes, i.e., both proportions fall in [α, $1 - \alpha$]
- With one node, both are 0.5

- With two, the proportions are 1/3 and 2/3