|  |  |
| --- | --- |
| **Data Structures & Algorithms**  Diploma in IT, CSF  Year 2 (2024/25) Semester 4 | **Week 13** |
| **0.5 Hour** |
| **Tutorial 9 – Trees** | |

1. Draw a **full** binary tree of height 3.

How many nodes are there in a **full** binary tree of height 3?

N = 2^(3+1) – 1 = 16-1 = 15

How many nodes are there in a **full** binary tree of height 10?

N = 2^(10+1) – 1 = 2048 – 1 = 2047

1. Draw 3 **complete** binary trees of height 3.

A black line drawing of a triangle

Description automatically generatedA black and white image of a triangle

Description automatically generated

A black and white drawing of a molecule

Description automatically generated

1. Draw any 2 binary trees of height 3 that are neither **full** nor **complete**

A black line drawing of a dog

Description automatically generated

A black and white drawing of a line

Description automatically generated

4. A binary search tree is to be created with the items given below.

55 28 87 32 90 16 45 79 66 37

Assuming the items are inserted in the order given (i.e. from left to right),

1. Draw the binary search tree after the items are inserted

Root: 55

Left subtree of 55: 28

28’s left child: 16

28’s right child: 32

32’s right child: 45

45’s left child: 37

Right subtree of 55: 87

87’s left child: 79

79’s left child: 66

87’s right child: 90

A number in a triangle

Description automatically generated

1. Is the binary search tree balanced

Not balanced

1. What is the output when the tree is traversed ***inorder*** ?

In order: 16, 28, 32, 37, 45, 55, 66, 79, 87, 90

1. What is the output when the tree is traversed ***preorder*** ?

Preorder: 55, 28, 16, 32, 45, 37, 87, 79, 66, 90

1. What is the output when the tree is traversed ***postorder*** ?

Post order: 16, 37, 45, 32, 28, 66, 79, 90, 87, 55

1. What is the output when the tree is traversed ***level order*** ?

Level-order: 55, 28, 87, 16, 32, 79, 90, 45, 66, 37

5. A binary search tree is to be created with the items given below.

55 28 87 32 90 16 45 79 66 30

Assuming the items are inserted in the order given (i.e. from left to right),

1. Draw the binary search tree after the items are inserted

**Root** = 55

**Left Subtree**

28 (left child = 16, right child = 32)

32 (left child = 30, right child = 45)

**Right Subtree**

87 (left child = 79, right child = 90)

79 (left child = 66)

A black text on a white background

Description automatically generated

1. Is the binary search tree balanced ?

Balanced

1. What is the height of the binary search tree ?

Height is 4

1. What is the worst case comparisons required to search for an item ?

O Log(2n) because the area of search is halved after each comparison

1. What is the result when 88 is inserted into the tree ?

Since 90 has no left child, 88 becomes left child of 90

1. What is the result when 16 is deleted from the tree ?

Under 28 left child, we remove 16

1. What is the result when 79 is deleted from the tree ?

79 is deleted, 79 has exactly one child, that node is replaced with its single child (66)

1. What is the result when 32 is deleted from the tree ?

Replace 32 with 45, remove old 45 node from right sub tree.

(left most child of right sub tree)

6. A binary search tree is to be created with the items given below.

16 28 30 32 45 55 66 79 87 90

Assuming the items are inserted in the order given (i.e. from left to right),

1. Draw the binary search tree after the items are inserted

A line of footprints in a line

Description automatically generated

1. Is the binary search tree balanced ?

No, each node has its left sub tree empty, only continuing chain in right subtree, hence skewed.

1. What is the height of the binary search tree ?

10 Nodes

1. What is the worst case comparisons required to search for an item ?

O n