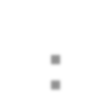
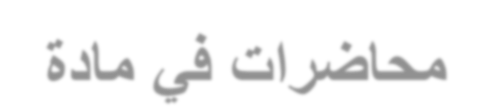
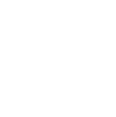
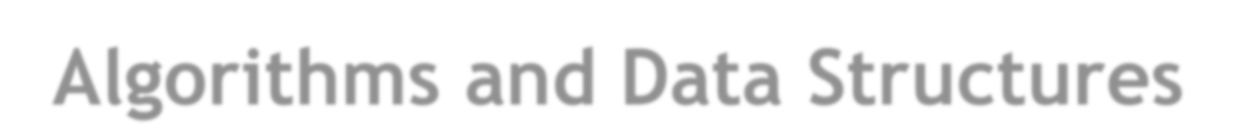
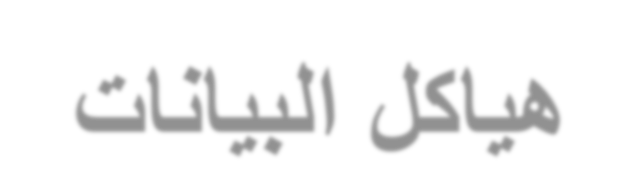
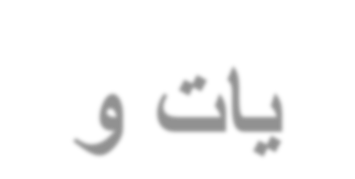
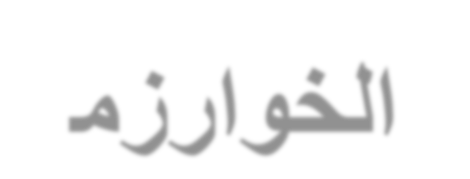
153



**Lecture # 7**

Trees

Outline

• Introduction to Trees

 Tree ADT.

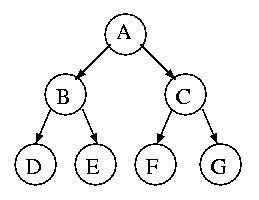
 Tree terminologies.

Trees

• Trees are data structures, that are composed of nodes organized by

references.

• The references are used to keep the nodes organized into a hierarchy.



Trees (cont…)

• A **tree** is a collection of nodes.

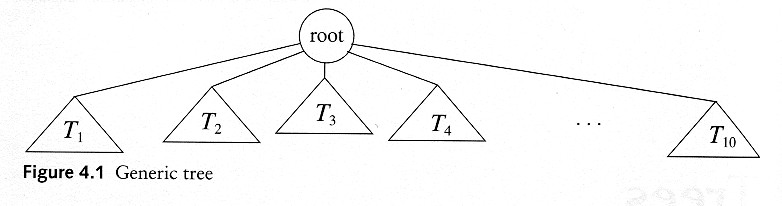
 The collection can be empty.

 **(recursive definition)** If not empty, a tree consists of:

 a distinguished node ***r*** (the ***root***).

 and zero or more nonempty *subtrees* **T1, T2, ... , Tk**.

• each of which are connected by a directed ***edge*** from ***r***.



Some Terminologies

• ***Root:***

 Node without parent.

• ***Child*** and ***Parent:***

 Every node except the root has one parent.

 A node can have zero or more children.

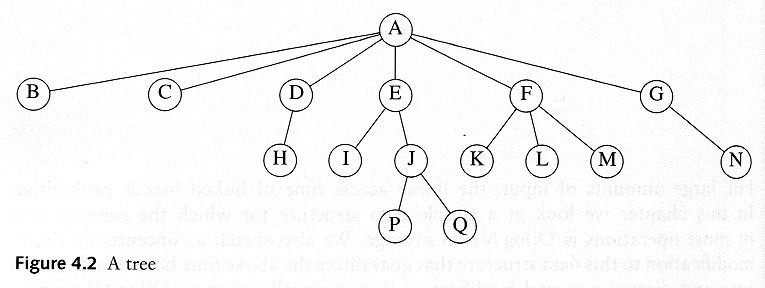
• ***Leaves:***

 Are nodes with no children.

• ***Siblings:***

 Nodes with same parent.

**Dr. Gasmelseed Ibrahim, International University of Africa, Faculty of Computer Studies** **Algorithms and Data Structures: Lecture (7)**



• ***Path:***

More Terminologies

 A sequence of edges.

• ***Length of a path:***

 Number of edges on the path.

• ***Depth of a node:***

 Length of the unique path from the root to that node.

 OR, number of ancestors of a node.

• ***Height of a node:***

 Length of the longest path from that node to a leaf.

 All leaves are at height **0**.

• ***The height of a tree*** = the height of the root

= the depth of the deepest leaf

More Terminologies (cont…)

• ***Ancestor*** and ***descendant:***

 If there is a path from **n**

**1**

to **n** :

**2**

 **1** is an *ancestor* of **n2**.

**n**

 **n** is a *descendant* of **n** .

**2 1**

 The ***ancestors*** of a node are: its parent, its grandparent, all the way to the root.

 The ***descendants*** of a node are: its children, their children, all the way down.

More Terminologies (cont…)

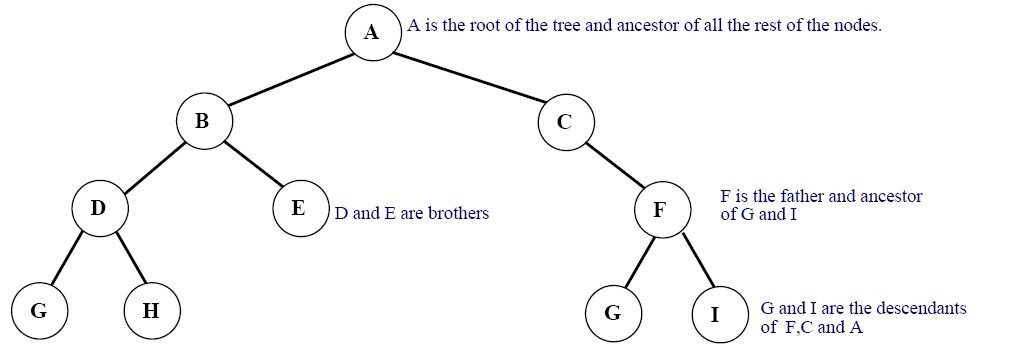
• The ***level*** of a node in a tree is defined as follows:

 The root of the tree has level **0**.

 The level of any other node in the tree is one more than the level of its father.

left son right son

(siblings)

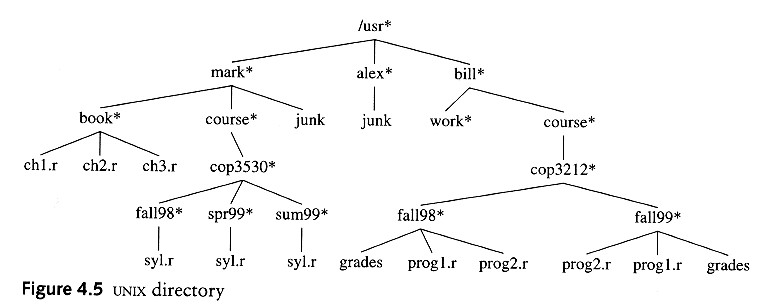


leaves

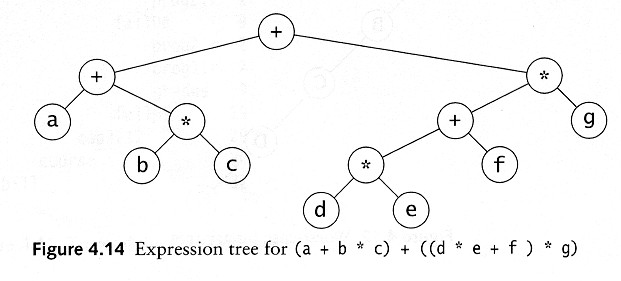


|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Depth of node **H** is: | **3** |  | Depth of node **A** is: | **0** |
|  | Depth of node **E** is: | **2** |  | Depth of this tree is: | **3** |

Example: UNIX Directory



Example: Expression Trees



 **Leaves** are **operands** (**constants** or **variables**).

 The **internal nodes** contain **operators**.

 Will not be a binary tree if some operators are not binary.