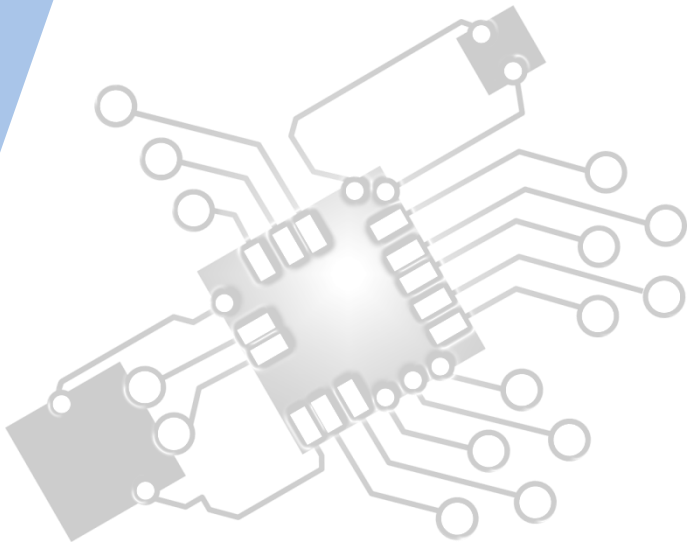




# *Planning & system installation*

IB Computer Science



*Content developed by  
Dartford Grammar School  
Computer Science Department*



# HL Topics 1-7, D1-4



1: System design



2: Computer Organisation



3: Networks



4: Computational thinking



5: Abstract data structures



6: Resource management



7: Control



D: OOP

# HL *only* 5 Overview

## Thinking recursively

- 5.1.1 Identify a situation that requires the use of recursive thinking
- 5.1.2 Identify recursive thinking in a specified problem solution
- 5.1.3 Trace a recursive algorithm to express a solution to a problem

## Abstract data structures

- 5.1.4 Describe the characteristics of a two-dimensional array
- 5.1.5 Construct algorithms using two-dimensional arrays
- 5.1.6 Describe the characteristics and applications of a stack
- 5.1.7 Construct algorithms using the access methods of a stack
- 5.1.8 Describe the characteristics and applications of a queue
- 5.1.9 Construct algorithms using the access methods of a queue
- 5.1.10 Explain the use of arrays as static stacks and queues

## Linked lists

- 5.1.11 Describe the features and characteristics of a dynamic data structure
- 5.1.12 Describe how linked lists operate logically
- 5.1.13 Sketch linked lists (single, double and circular)

## Trees

- 5.1.14 Describe how trees operate logically (both binary and non-binary)
- 5.1.15 Define the terms: parent, left-child, right-child, subtree, root and leaf
- 5.1.16 State the result of inorder, postorder and preorder tree traversal
- 5.1.17 Sketch binary trees

## Applications

- 5.1.18 Define the term dynamic data structure
- 5.1.19 Compare the use of static and dynamic data structures
- 5.1.20 Suggest a suitable structure for a given situation



1: System design

2: Computer Organisation



3: Networks

4: Computational thinking



5: Abstract data structures

6: Resource management



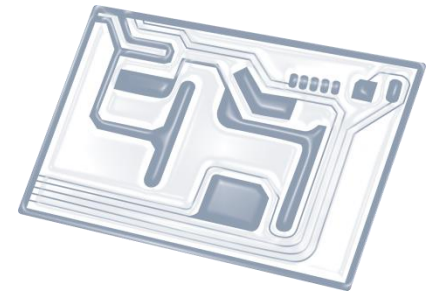
7: Control

D: OOP



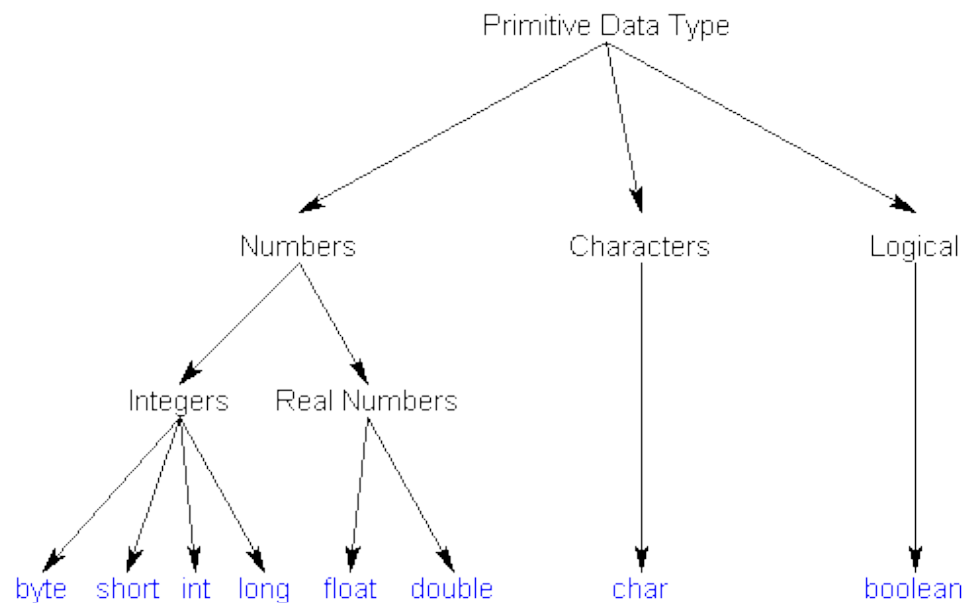
# Topic 5.1.17

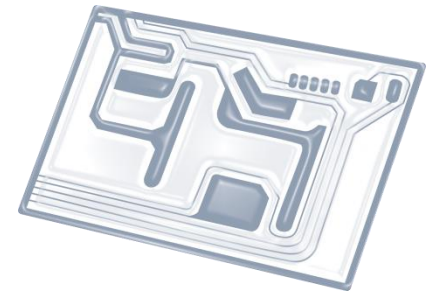
**Sketch** binary trees



# Abstract Data Structures (ADTs)

- 2D array
- Stack
- Queue
- Linked List
- **(Binary) Tree**
- Recursion





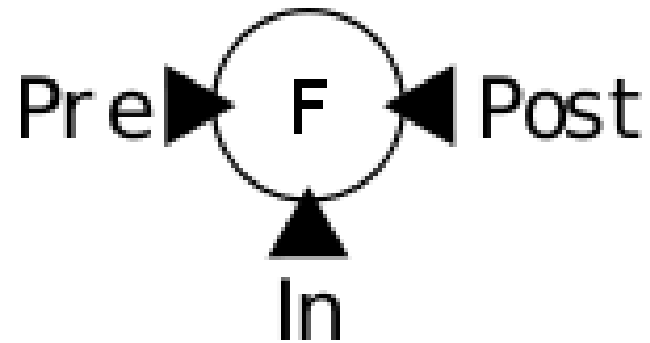
# Draw a tree for...

88, 25, 67, 97, 79, 99, 53, 87, 24, 76, 66, 32, 40, 42, 95

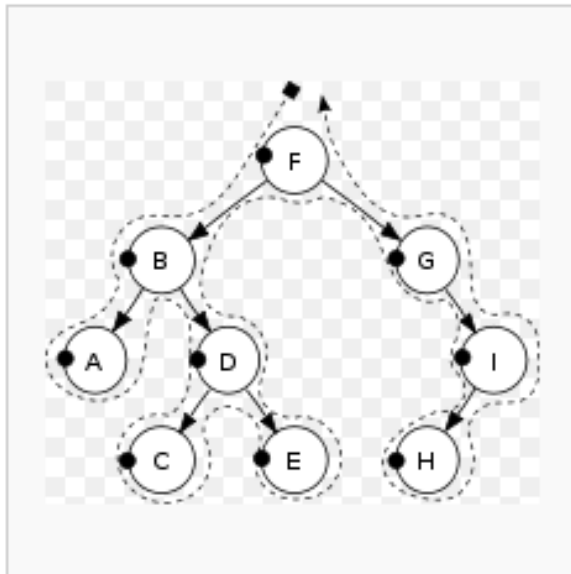
GH, WZ, AG, GM, IL, ZH, SV, RZ, HE, LZ, IJ

The names of the people in this class  
*(start with the people next to you...)*

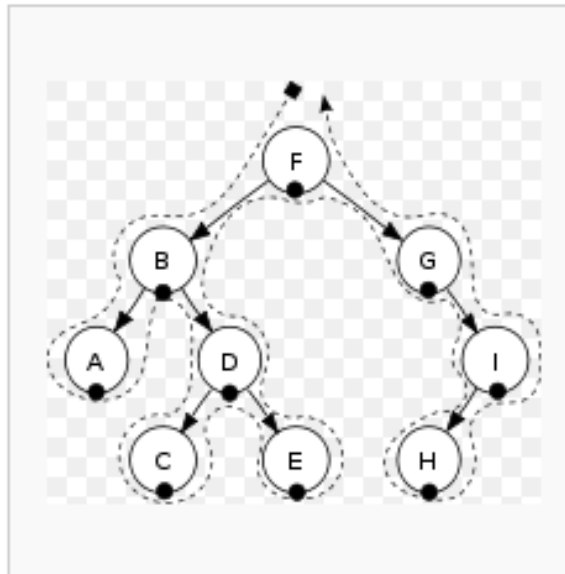
# Flag/Thread rule



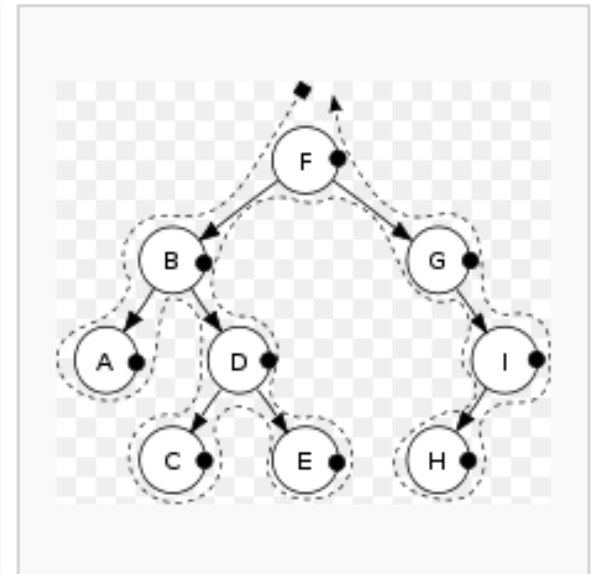
The 3 different types of traversal



Pre-order Traversal  
FBADCEGIH



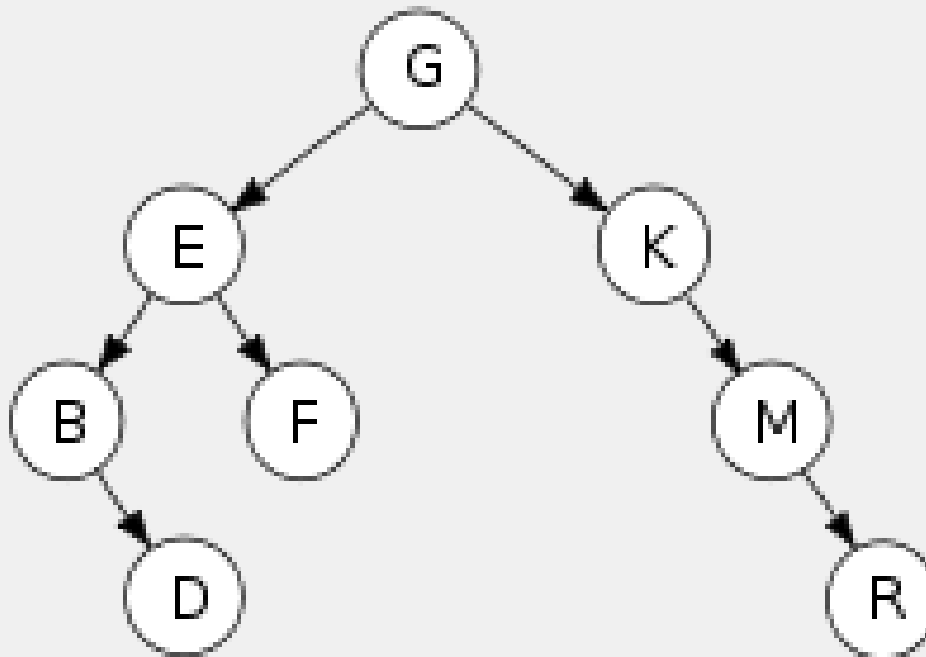
In-order Traversal  
ABCDEFGH I



Post-order Traversal  
ACEDBHIGF

# Question 1

For the following binary tree perform the following:

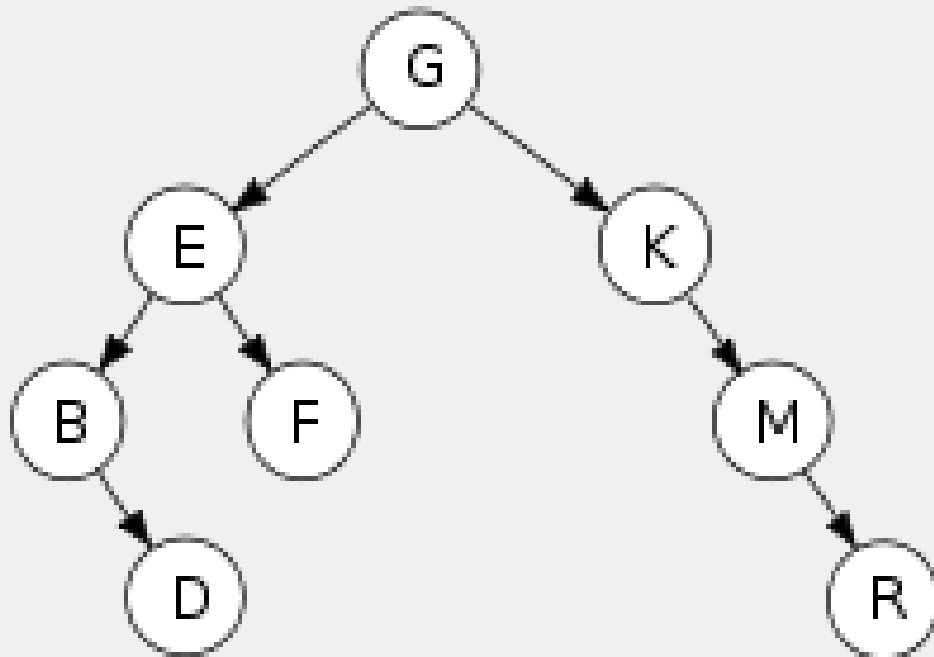


- Pre-order traversal
- In-order traversal
- Post-order traversal



# Answer 1

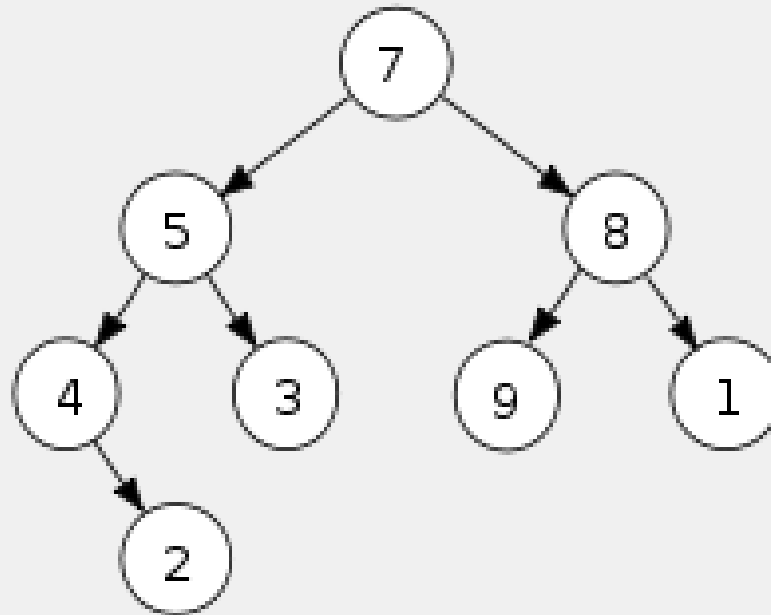
For the following binary tree perform the following:



- Pre-order traversal: GEBDFKMR
- In-order traversal: BDEFGKMR
- Post-order traversal: DBFERMKG

# Question 2

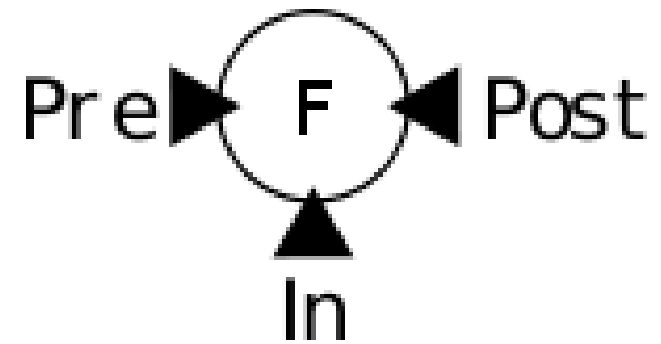
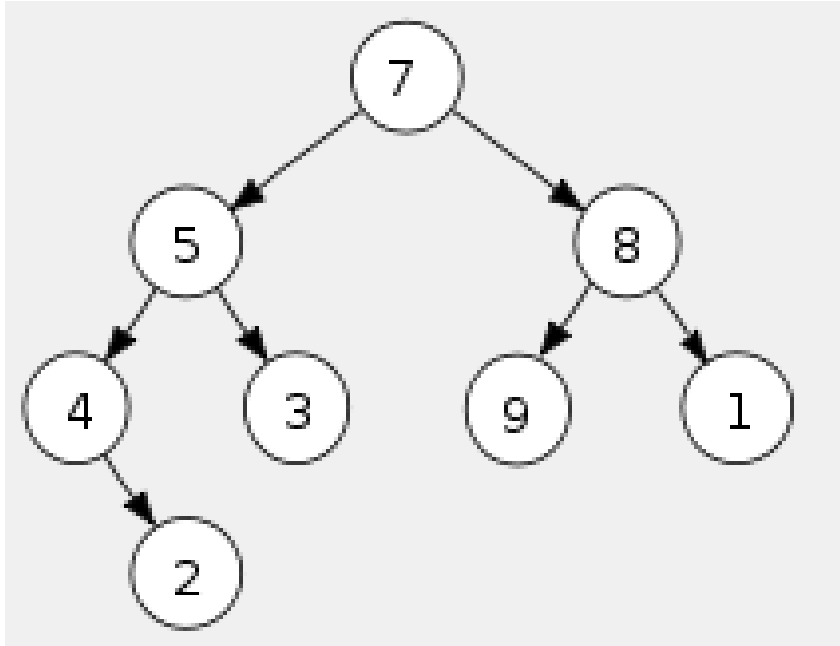
Using the following binary tree:



what would be the outputs for:

- Pre-order traversal
- In-order traversal
- Post-order traversal

# Answer 2

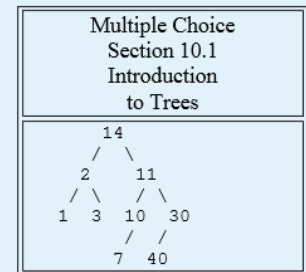


- **Pre-order** traversal: **7,5,4,2,3,8,9,1**
- **In-order** traversal: **4,2,5,3,7,9,8,1**
- **Post-order** traversal: **2,4,3,5,9,1,8,7**

# Useful practice questions (advanced)

## Multiple Choice

- There is a tree in the box at the top of this section. How many leaves does it have?
  - A. 2
  - B. 4
  - C. 6
  - D. 8
  - E. 9
- There is a tree in the box at the top of this section. How many of the nodes have at least one sibling?
  - A. 5
  - B. 6
  - C. 7
  - D. 8
  - E. 9
- There is a tree in the box at the top of this section. What is the value stored in the parent node of the node containing 30?
  - A. 10
  - B. 11
  - C. 14
  - D. 40
  - E. None of the above



Link: <http://www.cs.colorado.edu/~main/questions/chap10q.html>