Question 1

(a) Describe what is "**time complexity"** of an algorithm.
(b) Mathematically define the following asymptotic function notations.
   i. O (Big-O)
   ii. Ω (Big-Omega)
   iii. Θ (Theta)

(c) Assume that the following arithmetic expression is evaluated by using **Dijkstra's two-stack algorithm**.

2+8/4+3

   i. Illustrate the states of the **two stacks** after each step of the algorithm.
   ii. What is the time complexity of the algorithm?

Question 2

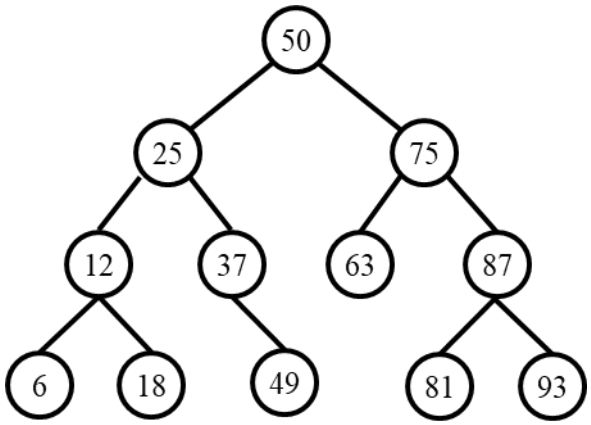

Figure 2.1 Binary Tree

(a) Answer the following questions considering the tree given in *Figure 2.1*.

   i. Is it a sorted binary tree?
   ii. Is it balanced?
   iii. What is the depth of the tree?
   iv. What is the size of the tree?
   v. Is the node 25 an ancestor of node 12?

(b) List the nodes in the order of visited if the tree in *Figure 2.1* is traversed in the following methods.

   i. Inorder
   ii. Postorder
   iii. Preorder

(c) Postorder is an ideal method to clone an entire binary tree.

   i. Briefly explain why postorder is suitable to clone a binary tree.
   ii. Write a **recursive** function in **Java** to print the values of a binary tree with **postorder-traversal**.

Note: *Assume that the nodes are instances of a Java class "**BinaryTree**" and the children are accessible through the members "**leftChild"** and "**righChild"** of **BinaryTree**.*

Question 3

7	2	8	5	4

Figure 3.2 An array of 5 elements

(a) Show how the values in the array shown in *Figure 3.1* will be positioned after applying the **first two steps** of each of the following sorting algorithms.

i. Bubble sort.
ii. Selection sort.
iii. Insertion sort

(b) Show how the **selection sort** is computationally cheaper than **bubble sort** in the **average case**, regardless of the both are having the same time complexity.

(c) **Heapsort** and **Quicksort** are typically used in time-critical applications.

i. What is **heap**, which is used in heapsort.
ii. Compare the **time complexities** of the two sorting algorithms in both the **average** and **worst** cases.
iii. Illustrate the states in a **single** iteration of heapsort algorithm using the heap given in *Figure 3.2*.

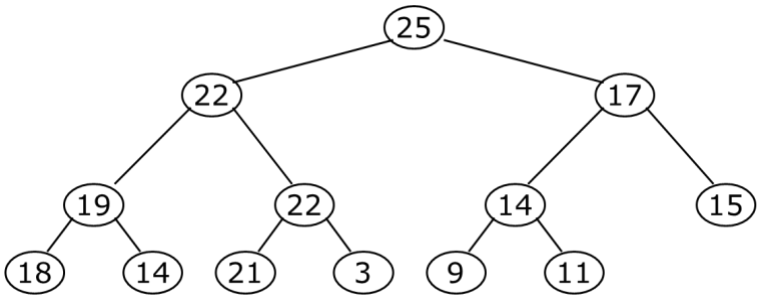

Figure 3.2 Heap of integers

Question 4

(a) Briefly describe the importance of the **header node** in LinkedList data structure.

(b) Queue data structure can be implemented using a Singly Linked List (SLL).

i. Suggest a method to implement a queue using SLL with enqueuing and dequeuing nodes in **O(1)** time complexity.

ii. Write the Java code of **enqueue** and **dequeue** methods of your suggestion, assuming the nodes are instances of a Java class **"Node".**

(c) Hashing is a technique which maps an input to a numerical value.

i. What is the goal of any hashing algorithm?
ii. What is **"collision"** in hashing?
iii. List **two** advantages of using bucket hashing over open hashing methods.