International University - VNUHCM
School of Computer Science and Engineering

Final Examination

Date: 29/01/2024; Duration: 120 minutes

Open-book and NO electronic devices

SUBJECT: Algorithms & Data Structures (IT013IU)	
Approval by The SCSE Signature Full name: Trần Thanh Tùng	Lecturer: Signature Full name: Vi Chí Thanh
Proctor 1 Signature Full name:	Proctor 2 Signature Full name:
STUDENT INFO	
Student name: **Student ID:**	

INSTRUCTIONS: the total point is 100 (equivalent to 40% of the course)

Purpose:

- Test your knowledge on data structures and algorithms in the following topics: Binary Tree, Hash Table, Graphs, Advanced graph algorithms
- Examine your skill in analysis and design algorithms

Student Name:.....
Student ID:.....

1. **Binary search tree (30 marks)**

Given a list of items, take items one by one from left to right.

40	73	6	24	33	29	50	45	12	67	52	15	18	24	67	9

1.a. Insert all items into a binary search tree and draw the tree (15 marks)

1.b. Delete the root node and redraw the tree (15 marks)

2. **Graph - Elementary Algorithms (20 marks)**

Find the topological sort order of all nodes in the graph G_S.

Show the order

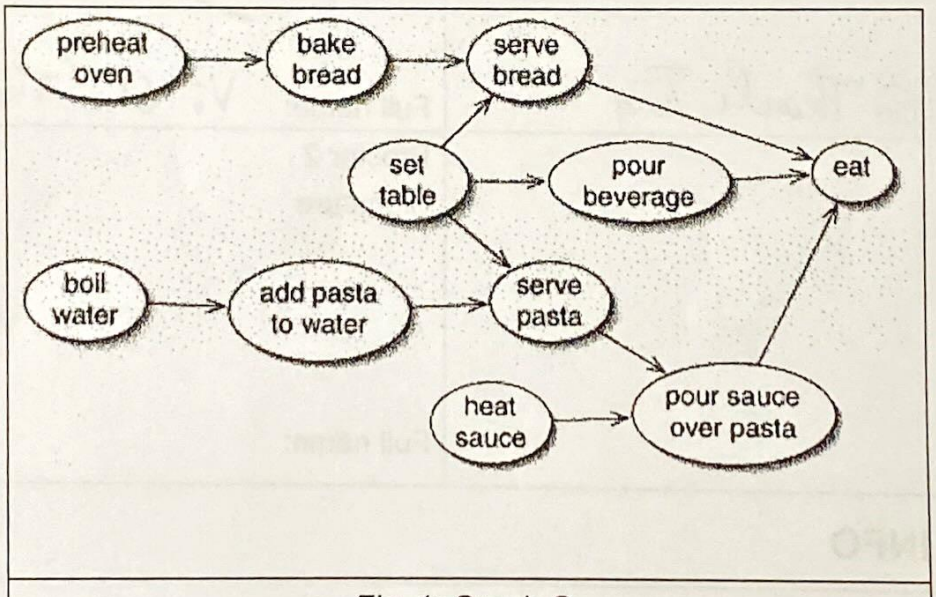

Fig. 1: Graph G_S

3. **Search and Print Leaves in a Binary Search Tree with Insertion**

You are given the definition of a binary search tree (BST) node as follows:

```
class TreeNode {
    int data;
    TreeNode left, right;

    public TreeNode(int data) {
        this.data = data;
        this.left = this.right = null;
    }
}
```

Student Name:.....
Student ID:.....

Your task is to implement the following Java methods:

a. (20 marks) `insert(TreeNode root, int value)` - Takes the root of a binary search tree and an integer value as arguments and inserts a new node with the given value into the tree. Returns the root of the modified binary search tree after insertion.

b. (20 marks) `findAndPrintDivisibleByFive(TreeNode root)` - Takes the root of a binary search tree and prints all nodes in the tree whose values are divisible by 5.

c. (10 marks) `printLeaves(TreeNode root)` - Takes the root of a binary search tree where each node has an integer value and prints the values of leaves in the tree.

Function Signatures:

```
public static TreeNode insert(TreeNode root, int value)

public static void printLeaves(TreeNode root)

public static void findAndPrintDivisibleByFive(TreeNode root)
```

Example:

Input:

```
TreeNode root = new TreeNode(10);
insert(root, 5);
insert(root, 15);
insert(root, 3);
insert(root, 7);
insert(root, 12);
insert(root, 18);

printLeaves(root);
findAndPrintDivisibleByFive(root)
```

Output:

```
Nodes divisible by 5: 5 10 15
Leaf Values: 3 7 12 18
```

--- The end ---