UNIVERSITY OF COLOMBO, SRI LANKA

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY

Academic Year 2012/2013 –2nd Year Examination – Semester 4

IT4104: Programming II

Part 1: Multiple Choice Question Paper

20st July, 2013
(ONE HOUR)

Important Instructions :

- The duration of the paper is **1 (one) hour.**
- The medium of instruction and questions is English.
- The paper has **25 questions** and **7 pages.**
- All questions are of the MCQ (Multiple Choice Questions) type.
- All questions should be answered.
- Each question will have 5 (five) choices with **one or more** correct answers.
- All questions will carry equal marks.
- There will be a penalty for incorrect responses to discourage guessing.
- The mark given for a question will vary from 0 (*All the incorrect choices are marked & no correct choices are marked*) to +1 (*All the correct choices are marked & no incorrect choices are marked*).
- Answers should be marked on the special answer sheet provided.
- Note that questions appear on both sides of the paper.
  If a page is not printed, please inform the supervisor immediately.
- Mark the correct choices on the question paper first and then transfer them to the given answer sheet which will be machine marked. **Please completely read and follow the instructions given on the other side of the answer sheet before you shade your correct choices.**

1) Consider the following table having two columns. In Column **I** different searching techniques are written. In **Column 2** some big O notations are shown. Entries in the **Column 1** are not written to match entries in **Column 2.**

Column 1		**Column 2**	
A	Binary	I	O(n)
B	Hashing	II	O(lg n)
C	Sequential	III	O(1)

Match each entry from **Column I** with the most appropriate entry in **Column 2**.

(a) A → I, B → II, C → III	(b) A → III, B → II, C → I	(c) A → III, B → I, C → II
(d) A → I, B → III, C → II	(e) A → II, B → I, C → III	

2) Consider the following paragraph.

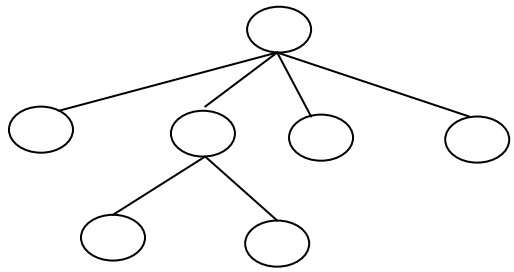
"In searching a value it is necessary to have a key to locate the value. As a different approach to searching, one can calculate the position or key of the index to store a particular value based on that value."

Select from among the following, (a) valid name/s that can be used for the process mentioned.

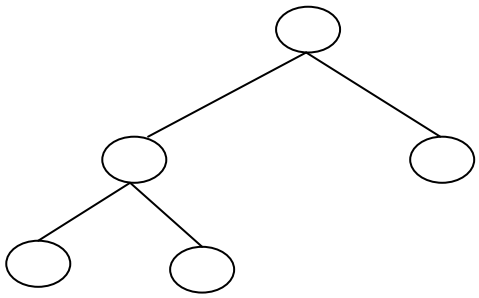
(a) Hashing	(b) Traversing	(c) Folding
(d) Sorting	(e) Addressing	

3) Consider the following diagrams which are numbered from I to IV.

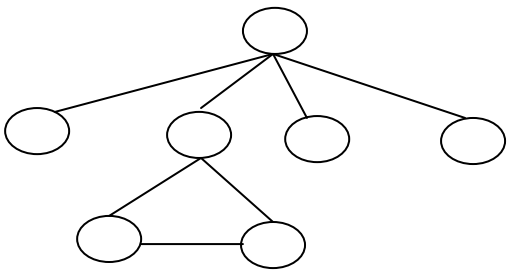
I



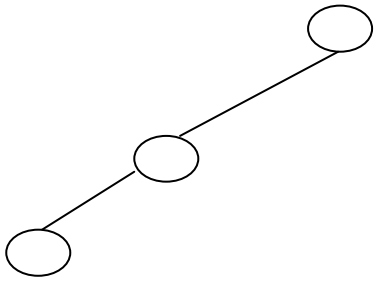
II



III



IV



Select from among the following, valid options that can be considered as Trees.

(a) I only.	(b) I, II and III only.	(c) I, III and IV only.
(d) I, II and IV only.	(e) III only.	

Consider the following program written in Java to answer question 4 – 5.

```
public class WhatCalculator {

    public static void main(String args[]) {

          int number = 4;

                    System.out.print(what(number));

    }



    public static int what(int number){
        if(number < 2)
            return 1;
        else
             return what(number-2) + what(number -1);
    }
}
```

4) What would the output of the program be?

(a) 3	(b) 4	(c) 5
(d) 3 2 1 1	(e) error	

5) Select from among the following, the programming concept/s which has (have) been used in the program.

(a) Multiple inheritance	(b) Polymorphism	(c) Inheritance
(d) Recursion	(e) Data hiding	

6) Consider the following two properties which are describing a special type of a data structure.

I. The value of each node of the data structure is less than or equal to the values stored in each of its children.
II. The data structure is perfectly balanced and the leaves in the last level are all in the left most positions.

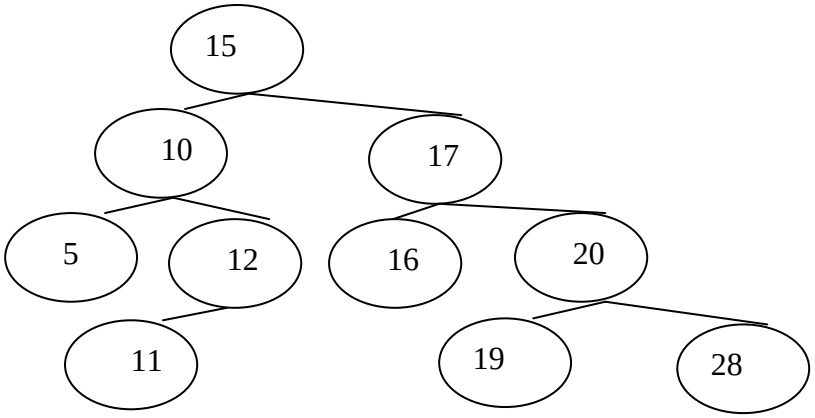
Select from among the following, a valid name which can be given to identify the special type of tree.

(a) Min heap	(b) Priority queue	(c) Tail Recursion
(d) Depth first traversal	(e) stack	

7) Select from among the following, the way/s one can implement a binary tree.

(a) an array	(b) a linked list	(c) a queue
(d) a graph	(e) a selection sort	

Consider the following tree data structure illustration to answer questions 8 to 13.



8) Select from among the following, the name which can be given for the node having the key 15.

(a) graph	(b) node	(c) root
(d) vertex	(e) arc	

9) Select from among the following, the number of leaf nodes in the tree.

(a) 6	(b) 5	(c) 4
(d) 3	(e) 2	

10) Select from among the following, the size of the tree.

(a) 9	(b) 10	(c) 8
(d) 7	(e) 6	

11) Select from among the following, the height of the given tree.

(a) 1	(b) 2	(c) 3
(d) 4	(e) 5	

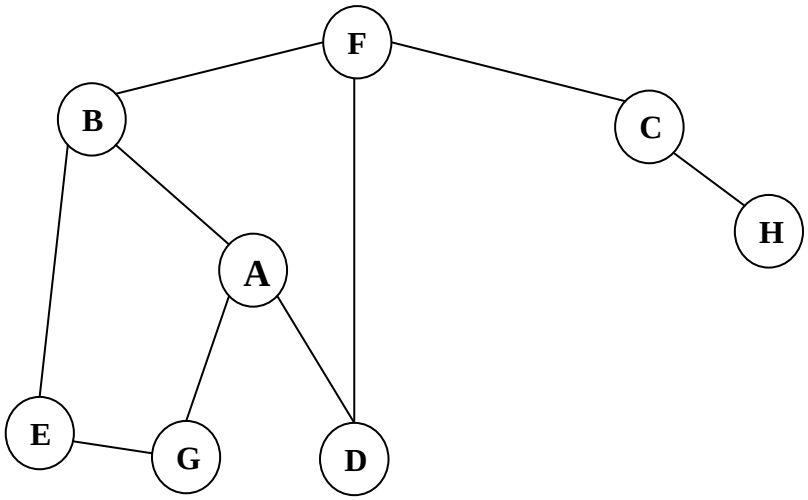
12) Select from among the following, the internal node/s that can be seen in the tree.

(a) 15	(b) 10	(c) 5
(d) 12	(e) 11	

13) Select from among the following, the ancestor/s of the node 16.

(a) 15	(b) 10	(c) 12
(d) 16	(e) 17	

Consider the following Graph illustration to answer the questions 14 and 15.



14) After applying a Graph traversal technique, the following result has been identified.

ABEGFCHD

Select from among the following, the Graph traversal technique that has been applied to obtain the above result.

(a) Depth-first	(b) Pre order	(c) Post order
(d) In order	(e) Breadth-first	

15) One has applied another Graph traversal technique and obtained the following result.

ABDGEFCH

Select from among the following, the Graph traversal technique that has been applied to obtain the above result.

(a) Depth-first	(b) Pre order	(c) Post order
(d) In order	(e) Breadth-first	

16) Consider the following pseudo code of a popular algorithm related to a Graph data structure.

```
whatMethod(weighted connected undirected graph)
    tree = null;
    edges = an unsorted sequence of all edges of  graph;
    for  j = 1 to | E |
      add e_j to tree;
      if there is a cycle in tree
         remove an edge with maximum weight from this only cycle;
```

Select from among the following, the purpose of the algorithm.

(a) Inserting a node	(b) Identifying a Spanning tree
(c) Identifying a shortest path	(d) Creating a directed graph
(e) Implementing the Kruskal's algorithm	

Consider the following declarations and initializations of variables to answer questions 17 to 23.

```
private int maxSize;
private long[] ex1Array;
private int top;

private int maxSize;
private long[] ex2Array;
private int front;
private int rear;
private int nItems;
```

And then consider the following names of a few data structures and their identifiers.

I Stack
II Queue
III Tree
IV Graph

Each question 17 to 23 is given a segment of code representing a function of a particular data structure as a method, written in Java. Method name is represented as a <u>blank</u>. Identify the functionality and then the name of that method to fill in the <u>blank</u> along with the identifier of the data structure as indicated in the above list.
E.g.
Method Name → identifier of the data structure (assuming the data structure is a Queue)
pop → II

17) public void **<u>blank</u>**(long j)
{ ex1Array[++top] = j; }

(a) push → I	(b) delete → III
(c) enqueue → II	(d) pop → I
(e) pop → IV	

18) public long **<u>blank</u>**()
{ return ex1Array[top--]; }

(a) push → I	(b) delete → III
(c) enqueue → II	(d) pop → I
(e) pop → IV	

19) public long **<u>blank</u>**()
{ return ex1Array[top]; }

(a) peek → I	(b) pop → III	(c) push → I
(d) isEmpty → I	(e) dequeue → II	

20) public boolean **<u>blank</u>**()
{ return (top == -1); }

(a) peek → I	(b) pop → III	(c) push → I
(d) isEmpty → I	(e) dequeue → II	

21) public void **blank**(long j) {
if(rear == maxSize-1)
rear = -1;
ex2Array[++rear] = j;
nItems++;
}

(a) push → I	(b) delete → III	(c) enqueue → II
(d) pop → I	(e) pop → IV	

22) public long **blank**()
{
long temp = ex2Array[front++];
if(front == maxSize)
front = 0;
nItems--;
return temp;
}

(a) peek → I	(b) pop → III	(c) isFull → II
(d) isEmpty → I	(e) dequeue → II	

23) public boolean **blank**()
{ return (nItems==maxSize); }

(a) peek → I	(b) pop → III	(c) isFull → II
(d) isEmpty → I	(e) dequeue → II	

24) Select from among the following, what can be considered as sorting algorithms.

(a) insertion	(b) selection	(c) radix
(d) folding	(e) division	

25) Select from among the following, what can be considered as a label setting algorithm.

(a) Dijkstra	(b) Selection	(c) Spanning tree
(d) Stackless depth first traversal	(e) Breadth first search	

****