



UNIVERSITY OF COLOMBO, SRI LANKA

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY (EXTERNAL)

*Academic Year 2013/2014 – 2<sup>nd</sup> Year Examination – Semester 4*

***IT4104: Programming II***  
***PART I- Multiple Choice Question Paper***

**19<sup>th</sup> July, 2014**  
**(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 **12 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 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. There are the three main measures of the running time of a program that can be applied to algorithms: best case, average case and worst case.

Consider the following table.

Section A	Section B
(i) Best Case	(A) This case occurs when the target is the last element in the list.
(ii) Average Case	(B) This case occurs when the target is the first element in the list.
(iii) Worst Case	(C) This case occurs when the target element is in the middle of the list.

Choose the best definition from section B for the three cases in section A.

- (a) (i) matches with (B) , (ii) matches with (A) and (iii) matches with (C)  
 (b) (i) matches with (B) , (ii) matches with (C) and (iii) matches with (A)  
 (c) (i) matches with (C) , (ii) matches with (B) and (iii) matches with (A)  
 (d) (i) matches with (A) , (ii) matches with (C) and (iii) matches with (B)  
 (e) (i) matches with (C) , (ii) matches with (A) and (iii) matches with (B)

2. Hand execute the following pseudo code segment and show the contents of the resulting queue.

```

values = Queue ()
for (int i=0;i<20;i++)
{
    if (i % 3 == 0 )
        values.enqueue( i )
    elseif (i % 4 == 0 )
        values.dequeue()
}

```

- (a) 0,3,6,9,12,15,18  
 (b) 3,9,12,15,18  
 (c) 9,12,15,18  
 (d) 6,9,12,15,18  
 (e) 0,9,12,18

3. Consider the following operations.

- (i) Reversing letters in a string, reversing words in a line or reversing a list of numbers  
 (ii) Find out whether a given string is a palindrome.  
 (iii) Arrange a list to store a waiting line of customers in a book store.  
 (iv) Creating a dictionary  
 (v) Examine a file to see if its braces { } and other operators match.

Which of the above operation(s) is/are implementable using stacks?

- (a) (i) (b) (ii) (c) (iii)  
 (d) (iv) (e) (v)

4. Consider the following pseudo code algorithm segment

```
Read (number)
Loop (number > 0)
    digit = number modulo 2
    print (digit)
    number = number / 2
```

If one uses 19 as the number, what will the output be?

- |           |           |           |
|-----------|-----------|-----------|
| (a) 10011 | (b) 11001 | (c) 10111 |
| (d) 10101 | (e) 00110 |           |

5. Consider the following pseudo code algorithm segment.

**BEGIN A:**

Initialize Stack **S1** to an Empty

Read a String from the Input Stream

While( Tokens remains in the Input String )

{

    Read a Token from the Input String

    If ( Token is a Number )

        Push it into the Stack **S1**

    Else If(Token is an Operator and Top two Numbers Exist in **S1**)

        Pop top two Numbers from the Stack **S1**

        Perform the indicated Operation

        Push the answer back into the Stack **S1**

    Else

        Error – invalid expression

}

Pop the answer off the Stack **S1**

**END**

If one evaluates the following postfix expression using the above algorithm, what would the final outcomes be from the given answers?

**Postfix expression is : 742+-382/+\***

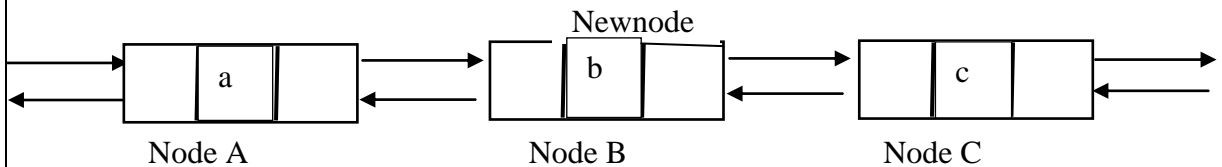
- |         |        |        |
|---------|--------|--------|
| (a) 7   | (b) 49 | (c) 52 |
| (d) 7,1 | (e) -7 |        |

6. Which of the following is/are **not** a valid queue application?

- |  |
|--|
| (a) When a series of printing jobs is submitted to a network printer |
| (b) Lines at ticket counters   |
| (c) Evaluating a mathematical expression                             |
| (d) If all lines are busy when making a call to large companies      |
| (e) Determining the shortest path from different airline paths       |

7. Consider the following doubly linked list and the java code segment.

Note : There are two references forward and backward



`Newnode.backward.forward=newnode.forward`

`Newnode.forward.backward=newnode.backward`

What is the above Java program segment intended to do?

- (a) Insert the Newnode into the end of the doubly linked list.
- (b) Insert Node B between Node A and Node C.
- (c) Delete Node A from the Doubly Linked List.
- (d) Delete Node B from the doubly linked List.
- (e) Combine Node A and Node C together.

8. Which of the following statement(s) is/are valid in connection with lists?

- (a) In a linked list, all cells are adjacently stored in the memory.
- (b) In a singly linked list, each cell contains the element(s) and a reference to a record containing its successor.
- (c) In an array based list, the maximum size of the array has to be declared at the time of declaration.
- (d) In an array based list, inserting a new element at position 0 requires first pushing the entire array down one position to make room.
- (e) In an array-based list, deleting elements from the middle can be performed without shifting the remaining elements.

9. Consider the following definition of the Fibonacci number series and the recursive pseudo code algorithm segment given thereafter.

**Definition :**

$$\text{fib}(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ \text{fib}(n-1) + \text{fib}(n-2) & \text{if } n \geq 2 \end{cases}$$

**Pseudo code algorithm:**

Begin

input: integer n such that  $n \geq 0$

if n is 0, return 0;

else if n is 1, return 1 ;

else return [ fib(n-1) + fib(n-2) ] ;

end

If a person hand executes the above pseudo code algorithm, what will the output be?

- (a) If n=1, return value is 1
- (b) If n=2, return value is 2
- (c) If n=8, return value is 21
- (d) If n=4, return value is 4
- (e) If n=6, return value is 8

10.

The following shows the base value of a function and its recursive format.

$f(1)=3$ ;

$f(n+1)=f(n)+3$

The appreciate recursive pseudo code algorithm for the above function and main part of the program segment is given below.

Function question9 (n):

if  $n == 1$

return 3;

else

return question9(n-1) + 3;

for  $i = 1$  to 10

print(question9(i));

If a person hand executes the above program segments, what would the output be?

(a) 3,6,9

(b) 3,6,9,12,15,18,21,24,27,30

(c) 3,6,9,12,15,18,21,24,2

(d) 3

(e) 0,3,6,9

11.

Consider the following recursive pseudo code algorithm.

Function question10(n):

if  $n == 0$ :

return 0

else:

return  $n + \text{question10}(n-1)$

print (question10(10))

If a person hand executes the above pseudo code algorithm, what would the output be?

(a) 10

(b) 9

(c) 53

(d) 55

(e) 45

12.

Which of the following statement(s) is/are correct in relation to AVL Trees?

(a) If three nodes lie in a straight line, a double rotation is needed to restore the balance.

(b) If three nodes lie in a straight line, a single rotation is needed to restore the balance.

(c) If three nodes lie in a dog-leg pattern (that is, there is a bend in the path), you need to perform a double rotation to restore the balance.

(d) If three nodes lie in a dog-leg pattern (that is, there is a bend in the path), you need to perform a single rotation twice in opposite directions to restore the balance.

(e) A single rotation involves always in shifting the middle node up to replace the top node and the top node becomes the left child of the middle node.

13.

Consider the set of integers

{30, 25, 36, 33, 35, 48, and 45}.

Create a Binary Search Tree using the above set of integers and store the node's content in an array. After deleting the node number 36 of the created binary search tree, what would the array index and contents of the resulting tree be?

(a)	Index	1	2	4	5	9	11
	Content	30	25	33	35	45	48
(b)	Index	1	2	3	6	7	14
	Content	30	25	35	33	48	45
(c)	Index	1	2	3	6	7	13
	Content	30	25	45	33	48	35
(d)	Index	1	2	3	6	7	14
	Content	30	25	33	35	48	45
(e)	Index	1	2	3	4	6	6
	Content	30	25	33	35	48	45

14.

Consider the following statements.

- (i) A binary tree can contain at most  $2^L$  nodes at Level L.
- (ii) The total number of nodes ( $T_n$ ) can contain  $2^{d+1} + 1$  nodes (d denotes the depth of the tree).
- (iii) One use of the binary tree is in the expression tree which is a central data structure in a compiler design.
- (iv) In a binary search tree, the root value is always higher with respect to the values in the left sub tree and values in the right sub tree are greater with respect to the root value.

Which of the following is/are correct in respect to the above statement regarding the trees?

(a) (i), (ii) and (iv) only

(b) (i), (ii) and (iii) only

(c) (i), (iii) and (iv) only

(d) (ii), (iii) and (iv) only

(e) All of these

15 Consider the following binary trees.

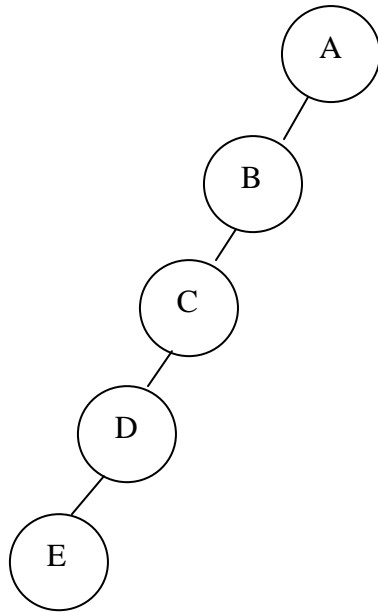


Figure (i)

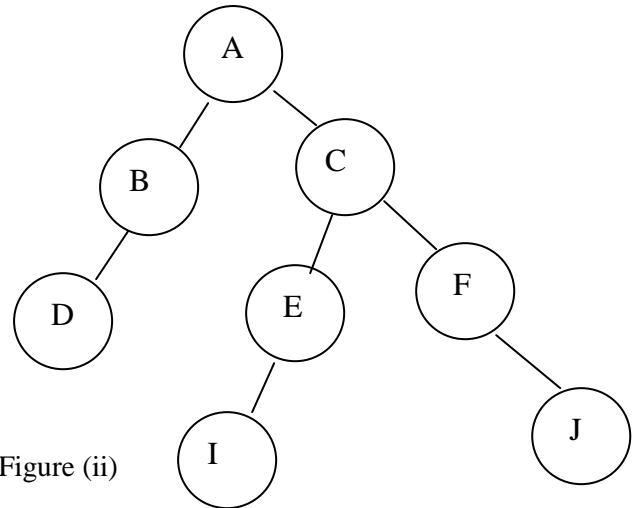


Figure (ii)

What is the order of nodes of the trees if traversed in the following sequence?

Pre-order

In-order

Post-order

- |     |             |   |
|-----|-------------|---|
| (a) | Figure (i)  | (A,B,C,D,E),(E,D,C,B,A),(E,D,C,B,A)                   |
|     | Figure (ii) | (A,B,D,C,E,I,F,J) (D,A,B,I,E,C,F,J),(D,B,I,E,J,F,C,A) |
| (b) | Figure (i)  | (A,B,C,D,E),(E,D,C,B,A),(E,D,C,B,A)                   |
|     | Figure (ii) | (A,B,D,C,E,I,J,F) (D,B,A,I,E,C,F,J),(D,B,I,E,J,F,C,A) |
| (c) | Figure (i)  | (A,B,C,D,E),(E,D,C,B,A),(E,D,C,A,B)                   |
|     | Figure (ii) | (A,B,D,C,E,I,F,J) (D,B,A,I,E,C,F,J),(D,B,I,E,J,F,C,A) |
| (d) | Figure (i)  | (A,B,C,D,E),(E,D,C,B,A),(E,D,C,B,A)                   |
|     | Figure (ii) | (A,B,D,C,E,I,F,J) (D,B,A,I,E,C,F,J),(D,B,I,E,J,F,C,A) |
| (e) | Figure (i)  | (A,B,C,D,E),(E,D,C,B,A),(E,D,C,B,A)                   |
|     | Figure (ii) | (A,B,D,C,E,I,F,J) (D,B,A,C,E,I,F,J),(D,B,I,E,J,F,C,A) |

16. Consider the following Pseudo code algorithm.

Input: binary search tree T, node v, element e

Output:

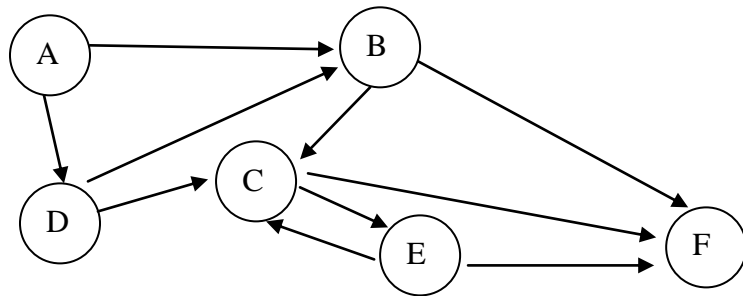
```
add(T, v, e){
    if(T.isLeaf(v)){
        if(v.element()>=e)
            add element e as v's left child
        else
            add element e as v's right child
    } else {
        if(v.element()>=e)
            add(T, T.leftChild(v), e)
        else
            add(T, T.rightChild(v), e)
    }
}
```

Identify, which of the following is/are true for the above program segment.

- (a) It covers how to insert new nodes into an empty binary search tree.
- (b) It also covers how to insert a new node into a binary search tree with only a root node.
- (c) It covers how to insert a new node to the binary search tree if the tree consists of left and right sub trees.
- (d) Adding a new item to a AVL tree
- (e) Adding a new item to a binary tree



Note : question No. 17 and 18 are based on the following directed graph



17 The Adjacency matrix of the above directed graph is

(a)

	A	B	C	D	E	F
A	F	T	F	T	F	F
B	F	F	T	F	F	F
C	F	F	F	F	T	F
D	F	T	T	F	F	F
E	F	F	T	F	F	T
F	F	F	F	F	F	F

(b)

	A	B	C	D	E	F
A	F	T	F	T	F	F
B	F	F	T	F	F	T
C	F	F	F	F	T	T
D	F	T	T	F	F	F
E	F	F	T	F	F	T
F	F	F	F	F	F	F

(c)

	A	B	C	D	E	F
A	F	T	F	T	F	F
B	F	F	T	T	F	F
C	F	F	F	F	T	F
D	F	T	T	F	F	F
E	F	F	T	F	F	F
F	F	F	F	T	F	F

d)

	A	B	C	D	E	F
A	F	T	F	T	F	F
B	F	F	T	F	F	F
C	F	F	F	F	T	F
D	F	T	F	F	T	F
E	F	F	T	F	F	T
F	F	F	F	F	F	F

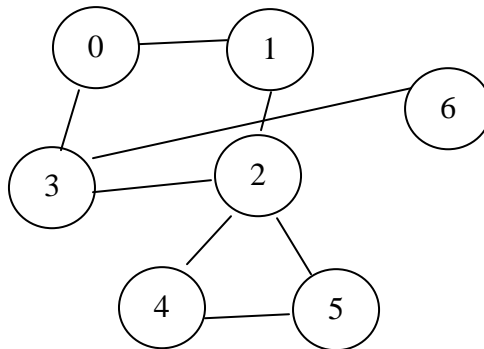
(e)

	A	B	C	D	E	F
A	F	T	F	T	F	F
B	F	F	T	F	T	F
C	F	T	F	T	F	F
D	F	F	F	F	F	T
E	F	T	F	T	T	F
F	F	F	F	F	F	F

18. Path Matrix (transitive closure) of the above directed graph is

(a)	(b)																																																																																																		
<table><tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>A</td><td>T</td><td>T</td><td>T</td><td>T</td><td>T</td><td>T</td></tr><tr><td>B</td><td>F</td><td>F</td><td>T</td><td>T</td><td>F</td><td>T</td></tr><tr><td>C</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>D</td><td>F</td><td>T</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>E</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table>		A	B	C	D	E	F	A	T	T	T	T	T	T	B	F	F	T	T	F	T	C	F	F	T	F	T	T	D	F	T	T	F	T	T	E	F	F	T	F	T	T	F	F	F	F	F	F	F	<table><tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>A</td><td>F</td><td>T</td><td>T</td><td>T</td><td>T</td><td>T</td></tr><tr><td>B</td><td>F</td><td>F</td><td>T</td><td>T</td><td>T</td><td>T</td></tr><tr><td>C</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>D</td><td>F</td><td>T</td><td>T</td><td>T</td><td>T</td><td>T</td></tr><tr><td>E</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr><tr><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table>		A	B	C	D	E	F	A	F	T	T	T	T	T	B	F	F	T	T	T	T	C	F	F	T	F	T	T	D	F	T	T	T	T	T	E	F	F	F	F	F	F	F	F	F	F	F	F	F
	A	B	C	D	E	F																																																																																													
A	T	T	T	T	T	T																																																																																													
B	F	F	T	T	F	T																																																																																													
C	F	F	T	F	T	T																																																																																													
D	F	T	T	F	T	T																																																																																													
E	F	F	T	F	T	T																																																																																													
F	F	F	F	F	F	F																																																																																													
	A	B	C	D	E	F																																																																																													
A	F	T	T	T	T	T																																																																																													
B	F	F	T	T	T	T																																																																																													
C	F	F	T	F	T	T																																																																																													
D	F	T	T	T	T	T																																																																																													
E	F	F	F	F	F	F																																																																																													
F	F	F	F	F	F	F																																																																																													
(c)	(d)																																																																																																		
<table><tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>A</td><td>F</td><td>T</td><td>T</td><td>T</td><td>T</td><td>T</td></tr><tr><td>B</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>C</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>D</td><td>F</td><td>T</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>E</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table>		A	B	C	D	E	F	A	F	T	T	T	T	T	B	F	F	T	F	T	T	C	F	F	T	F	T	T	D	F	T	T	F	T	T	E	F	F	T	F	T	T	F	F	F	F	F	F	F	<table><tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>A</td><td>F</td><td>T</td><td>F</td><td>T</td><td>F</td><td>F</td></tr><tr><td>B</td><td>F</td><td>F</td><td>T</td><td>F</td><td>F</td><td>F</td></tr><tr><td>C</td><td>F</td><td>F</td><td>F</td><td>F</td><td>T</td><td>F</td></tr><tr><td>D</td><td>F</td><td>T</td><td>F</td><td>F</td><td>T</td><td>F</td></tr><tr><td>E</td><td>F</td><td>F</td><td>T</td><td>F</td><td>F</td><td>T</td></tr><tr><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table>		A	B	C	D	E	F	A	F	T	F	T	F	F	B	F	F	T	F	F	F	C	F	F	F	F	T	F	D	F	T	F	F	T	F	E	F	F	T	F	F	T	F	F	F	F	F	F	F
	A	B	C	D	E	F																																																																																													
A	F	T	T	T	T	T																																																																																													
B	F	F	T	F	T	T																																																																																													
C	F	F	T	F	T	T																																																																																													
D	F	T	T	F	T	T																																																																																													
E	F	F	T	F	T	T																																																																																													
F	F	F	F	F	F	F																																																																																													
	A	B	C	D	E	F																																																																																													
A	F	T	F	T	F	F																																																																																													
B	F	F	T	F	F	F																																																																																													
C	F	F	F	F	T	F																																																																																													
D	F	T	F	F	T	F																																																																																													
E	F	F	T	F	F	T																																																																																													
F	F	F	F	F	F	F																																																																																													
(e)																																																																																																			
<table><tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>A</td><td>F</td><td>T</td><td>T</td><td>T</td><td>T</td><td>F</td></tr><tr><td>B</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>C</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>D</td><td>T</td><td>F</td><td>F</td><td>F</td><td>T</td><td>T</td></tr><tr><td>E</td><td>F</td><td>F</td><td>T</td><td>F</td><td>T</td><td>T</td></tr><tr><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table>		A	B	C	D	E	F	A	F	T	T	T	T	F	B	F	F	T	F	T	T	C	F	F	T	F	T	T	D	T	F	F	F	T	T	E	F	F	T	F	T	T	F	F	F	F	F	F	F																																																		
	A	B	C	D	E	F																																																																																													
A	F	T	T	T	T	F																																																																																													
B	F	F	T	F	T	T																																																																																													
C	F	F	T	F	T	T																																																																																													
D	T	F	F	F	T	T																																																																																													
E	F	F	T	F	T	T																																																																																													
F	F	F	F	F	F	F																																																																																													

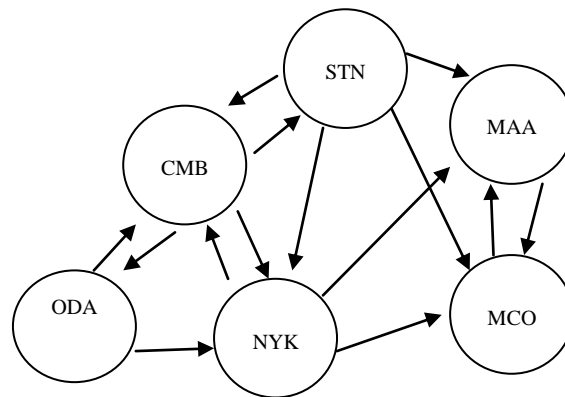
19. Consider the following un-directed graph.



If one traverses the above graph according to the Depth First Traversal (DFT), what would the order of nodes be?

- |   |   |
|---|---|
| (a) $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 4 \rightarrow 5$ | (b) $0 \rightarrow 1 \rightarrow 6 \rightarrow 3 \rightarrow 2 \rightarrow 4 \rightarrow 5$ |
| (c) $0 \rightarrow 1 \rightarrow 5 \rightarrow 4 \rightarrow 2 \rightarrow 6 \rightarrow 3$ | (d) $0 \rightarrow 3 \rightarrow 6 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 1$ |
| (e) $0 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 4 \rightarrow 5 \rightarrow 6$ |   |

20. Consider that the following directed graph which shows the names of air ports, denoted by Graph P.



Graph P

If one wants to go from CMB to MCO, what would be the possibilities of the order of air ports if one traverses in Breadth First Traversal (BFT)?

- (a) CMB→STN→ODA→NYK→MAA→MCO
- (b) CMB→NYK→ODA→STN→MCO→MAA
- (c) CMB→NYK→MCO→MAA→STN→ODA
- (d) CMB→ODA→NYK→STN→MCO→MAA
- (e) CMB→STN→MAA→MCO→STN→ODA

21. Let A be a list of n numbers. Sorting A refers to the operation of re-arranging the elements of A so that they are in ascending order.

If A= {35, 57, 24, 90, 67, 20, 9, 4} and method of sorting is bubble sort, which are the **correct** intermediate output values during the process of sorting?

- |                           |                           |
|---------------------------|---------------------------|
| (a) 35,57,24,90,4,67,20,9 | (b) 4,35,57,24,90,67,20,9 |
| (c) 35,24,57,90,67,20,9,4 | (d) 35,24,57,67,20,9,4,90 |
| (e) 57,35,4,24,90,67,20,9 |                           |

22. Consider the set of integer values given below.  
{52, 21, 11, 45, 85, 91, 8, 55, 49, 51}

If one chooses the first element as a pivot value, how many pivot values are required to sort the above set of integers completely using the quick sort algorithm?

- |       |       |       |
|-------|-------|-------|
| (a) 6 | (b) 7 | (c) 4 |
| (d) 5 | (e) 2 |       |

23. Consider the selection sort algorithm given below.

Selection\_Sort (data [])

for (i=0; i<data.length -1; i++)

{Select the smallest element among data ([i] .....data [data.length-1];

Swap it with data[i];}

If the data set is {50, 89, 16, 04, 75, 23, 96, 02} what would be the **incorrect** intermediate output values during the processes of sorting using the algorithm given above?

(a) {02,04,16,89,75,23,96,50}

(b) {02,04,16,89,75,23,96,50}

(c) {02,04,16,23,50,96,75,89}

(d) {02,04,16,23,89,75,96,50}

(e) {02,04,16,23,75,89,96,50}

24. Consider the following binary search pseudo code algorithm given below.

Set top=arraysize - 1; set bottom = 0;

While top > bottom

Set middle = (top+bottom)/2

If the key at index middle is equal to the target, return the middle.

Else if key at index middle is less than target key , then bottom=middle+1

Else set top=middle -1

Throw an exception indicating target has not been found.

How many comparisons are required to search when the target value is equal to 7 using the binary search algorithm if one uses the following data set?

Data set is {7,9,11,13,15,37,61,98,102}

**Note:** you may assume bottom and top values are set to first and last index of the data set respectively and when calculating the middle, answer should be taken as the upper integer value. (e.g 3/2 is equal to 2 (not 1))

(a) 1

(b) 9

(c) 2

(d) 3

(e) 4

25. Which of the following statements is/are correct in relation to the shell sort?

Assume that the initial array is X[1],x[2],.....X[15].

(i) If the increment (k) is 5, then the first sub-file contains X[1],X[5],X[10],X[15].

(ii) If increment (k) is 4, the first sub-file contains X[1],X[5],X[9],X[13].

(iii) In general,  $i^{\text{th}}$  element of the  $j^{\text{th}}$  sub-file is  $X[(I-1)*K+J]$ .

(iv) In general,  $i^{\text{th}}$  element of the  $j^{\text{th}}$  sub-file is  $X[(I-2)*K+J]$ .

(v) In general,  $i^{\text{th}}$  element of the  $j^{\text{th}}$  sub-file is  $X[(I-1)*K*J]$ .

(a) (i) and (iii) only

(b) (ii) and (iii) only

(c) (ii) and (v) only

(d) (i) and (v) only

(e) (ii) and (iv) only

\*\*\*\*\*