



UNIVERSITY OF COLOMBO, SRI LANKA

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY (EXTERNAL)

Academic Year 2008/2009 – 2nd Year Examination – Semester 4

IT4103: Programming II
Part 1: Multiple Choice Question Paper

15th August, 2009
(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 **6 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) Select from among the following, the valid statement(s) on Java applications and applets.

- (a) Every Java application contains a class that defines a method called *main()*.
- (b) The name of the class having the *main()* method defined is used as the argument to the Java interpreter.
- (c) To execute the bytecode program in the .class file with the Java interpreter, one has to write the following command in the command prompt.

java MyProgram.class

- (d) One can execute an applet either within a Java 2 enabled web browser or by using the *appletviewer* after embedding the applet in a web page.
- (e) The following is an example to show the execution of an applet.

appletviewer MyApplet.htm

2) One can describe real world data as physical entities which can store data and have existence external to the computer. Select from among the following, which can be considered as real world data.

- (a) The personal file of Mr. Oshadha Ranasinghe
- (b) An inventory record having the information of some grocery items
- (c) Random access memory of the computer
- (d) Prescription generated for patients at the hospital.
- (e) Financial transaction record which has been done to pay money for daily maintenance

Consider the following program taking note of blanks 1 – 4 to answer questions 3 and 6

```
class Problem1
{
private int maxSize;
private char[] stackArray;
private int top;
public Problem1(int max)
{
maxSize = max;
stackArray = new char[maxSize];
top = -1;
}
public void push(char j) // put item on top of stack
{
blank1
}

public char pop() // take item from top of stack
{
blank2
}

public char peek() // peek at top of stack
{
blank3
}

public boolean isEmpty() // true if stack is empty
{
blank4
}
} // end class Problem1
```

3) Select from among the following, the correct statement to fill the blank indicated as **blank1** in the program.

- | | |
|-------------------------------------|-----------------------------|
| (a) return stackArray[top--]; | (b) return stackArray[top]; |
| (c) stackArray[++top] = j; | (d) return (top == -1); |
| (e) stackArray = new char[maxSize]; | |

- 4) Select from among the following, the correct statement to fill the blank indicated as **blank2** in the program.

(a) return stackArray[top--];	(b) return stackArray[top];
(c) stackArray[++top] = j;	(d) return (top == -1);
(e) stackArray = new char[maxSize];	

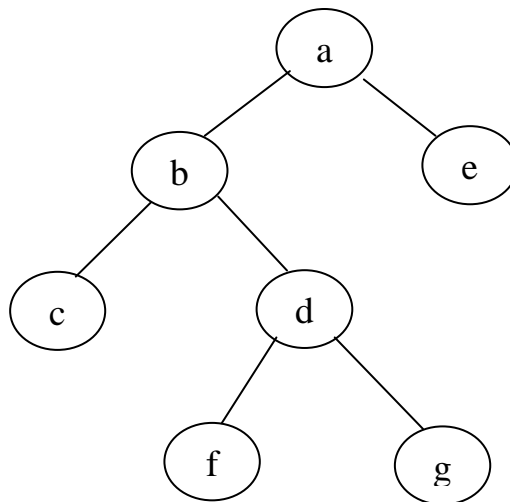
- 5) Select from among the following, the correct statement to fill the blank indicated as **blank3** in the program.

(a) return stackArray[top--];	(b) return stackArray[top];
(c) stackArray[++top] = j;	(d) return (top == -1);
(e) stackArray = new char[maxSize];	

- 6) Select from among the following, the correct statement to fill the blank indicated as **blank4** in the program.

(a) return stackArray[top--];	(b) return stackArray[top];
(c) stackArray[++top] = j;	(d) return (top == -1);
(e) stackArray = new char[maxSize];	

Consider the following tree structure to answer questions 7 – 9.



- 7) What would the output be, if the pre order traversal method is used to traverse the above tree?

(a) abcdefg	(b) abcdffe	(c) cbfdgae
(d) cfgdbea	(e) beacdfg	

- 8) What would the output be, if the in order traversal method is used to traverse the above tree?

(a) abcdefg	(b) abcdffe	(c) cbfdgae
(d) cfgdbea	(e) beacdfg	

- 9) What would the output be, if the post order traversal method is used to traverse the above tree?

(a) abcdefg	(b) abcdffe	(c) cbfdgae
(d) cfgdbea	(e) beacdfg	

10) Select from among the following, the correct type/s of implementing a recursion.

- | | | |
|------------|-------------|--------------|
| (a) tail | (b) radix | (c) indirect |
| (d) bubble | (e) nontail | |

Consider the following program written in Java to answer questions 11 and 12.

```
class What{
public static void what(int i){
    if(i > 0){
        what(i-1);
        System.out.print(i + "");
        what(i-1);
    }
}
public static void main(String args[]){
    What.what(2);
}}
```

11) Select from among the following, the correct recursive type which has been used in the program.

- | | | |
|------------|-------------|--------------|
| (a) tail | (b) radix | (c) indirect |
| (d) bubble | (e) nontail | |

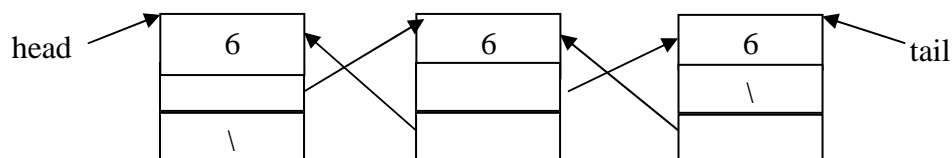
12) What would the output of the program be?

- | | | |
|---------|---------|--------|
| (a) 2 | (b) 3 | (c) 12 |
| (d) 121 | (e) 122 | |

13) Select from among the following, the correct statements on using skip lists.

- | |
|--|
| (a) A skip list makes nonsequential searching possible. |
| (b) Using skip lists, one can search for a data item in a circular manner. |
| (c) Since linked lists are allowing random access to data items, the usage of skip lists is essential to speed up the searching process. |
| (d) An array can be considered as a very powerful implementation of skip lists. |
| (e) Skip lists are advantageous compared to linked lists when it comes to look for a value in lists. |

14) Consider the following diagram illustrating a kind of a linked list data structure.



Select from among the following, the name which can be given for the above illustration.

- | | |
|---------------------------------|------------------------------|
| (a) Single Linked List | (b) Double Ended Linked List |
| (c) Circular Singly Linked List | (d) Doubly Linked List |
| (e) Skip List | |

15) Which of the following, are types of folding hash functions?

- | | | |
|--------------|--------------|----------------|
| (a) tail | (b) boundary | (c) mid-square |
| (d) division | (e) shift | |

- 16) Consider the following pseudocode to represent a searching algorithm which is used in Graph data structure.

```
ABC(v)
    num(v) = i++;
    for all vertices u adjacent to v
        if num(u) is 0
            attach edge(uv) to edges;
        ABC(U);

whatSearchMethod()
    for all vertices v
        num(v) = 0;
    edges = null;
    i = 1;
    while there is a vertex v such that num(v) is 0
        ABC(V);
    output edges;
```

Select from among the following, the searching method which is described by the pseudocode.

- | | | |
|-------------------|------------|---------------|
| (a) depth first | (b) bubble | (c) excessive |
| (d) breadth first | (e) shell | |

- 17) Select from among the following, the methods which can be used to solve the problem of calculating the shortest path in Graph data structure.

- | | |
|-------------------|------------------------------|
| (a) Stack | (b) Double Ended Linked List |
| (c) Label Setting | (d) Label Correcting |
| (e) Breadh First | |

- 18) Consider the following pseudocode which is written for solving a problem of sorting.

```
whatSort(data, first, last)
    if first < last
        mid = (first + last) / 2;
        whatSort(data, first, mid);
        whatSort(data, mid + 1, last);
        combine(data, first, last);
```

Select from among the following, the sorting method which is represented in the above pseudocode.

- | | | |
|------------|-----------|-----------|
| (a) merge | (b) shift | (c) shell |
| (d) bubble | (e) tail | |

- 19) At the top most level, one can identify two broad categories of events in Java. Which of the following are the two broad categories of events in Java?

- | | | |
|------------------------|--------------------|---------------------|
| (a) Low-Level events | (b) Mouse events | (c) Semantic events |
| (d) Middle-level event | (e) KeyTyped event | |

- 20) Select from among the following, basic methods which have been declared by the ResultSet interface.

- | | | |
|----------------------|---------------|---------------|
| (a) getAsciiStream() | (b) getInt() | (c) getDate() |
| (d) getShort() | (e) getTime() | |

- 21) Consider the following segment of code written in Java to solve a problem in sorting.

```
public void whatSort(object[] data){
    for(int i = 0; i < data.length-1; i++)
        for(int j = data.length-1; j > i; --j)
            if(((Comparable)data[j]).compareTo(data[j-1]) < 0)
                swap(data,j,j-1);
}
```

Select from among the following, the sorting method which is represented by the above code segment.

- | | | |
|-------------|-----------|--------------|
| (a) bubble | (b) shell | (c) non tail |
| (d) kruskal | (e) radix | |

- 22) Consider the following, segment of code written in Java to solve a problem in sorting.

```
public void whatSort(Object[] data){
    int i,j,least;
    for(i = 0; i < data.length-1; i++){
        for(j = i + 1, least = i; j < data.length ; j++){
            if(((Comparable)data[j]).compareTo(data[least]) < 0)
                least = j;
            if(i != least )
                swap(data,least,i);
        }
    }
}
```

Select from among the following, the sorting method which is represented by the above code segment.

- | | | |
|---------------|---------------|----------|
| (a) indirect | (b) insertion | (c) heap |
| (d) selection | (e) quick | |

- 23) Select from among the following, a suitable description for the method *getPath()* which is provided by the File class.

- | |
|---|
| (a) Returns a hash code value for the current File object |
| (b) Returns the parent directory as a File object, or <i>null</i> this File object does not have a parent |
| (c) Returns true if the File object refers to an existing directory and false otherwise |
| (d) Returns a String object containing the path for the file object including the file or directory name |
| (e) Returns the absolute path for the directory or file referenced by the current File object |

- 24) Select from among the following, (the) valid constructor(s) provided by the *FileOutputStream* class.

- | |
|---|
| (a) <i>FileOutputStream</i> (String fileName) |
| (b) <i>FileOutputStream</i> (int numberOfFiles) |
| (c) <i>FileOutputStream</i> (File file) |
| (d) <i>FileOutputStream</i> (File file, boolean append) |
| (e) <i>FileOutputStream</i> (char fileName) |

- 25) Select from among the following, what can be classified as sub classes of the class *Component*.

- | | | |
|------------|----------------|-----------|
| (a) Window | (b) Panel | (c) Frame |
| (d) Applet | (e) JComponent | |
