Question 1. Answer part (a) and *either* part (b) *or* part (c). [Total 65 marks]

Q1 Part (a)

The following text describes the organisation of conferences by an international organisation – IFIP. IFIP Policy on conferences suggest the appointment of a Programme Committee to deal with the technical content of the conference and an Organising Committee to handle financial matters, local arrangements, and invitations and publicity. These committees clearly need to work together closely and have a need for common information and to keep their recorded information consistent and up to date. The software system that is under consideration includes support for the activities of both a Programme Committee and an Organising Committee involved in arranging an IFIP Working Conference. The involvement of the two committees is seen as analogous to two organisational entities within a corporate structure using some common information. The following activities of the committees are significant.

Programme Committee

1. Preparing a list to whom the call for papers is to be sent.
2. Registering the letters of intent received in response to the call.
3. Registering the contributed papers on receipt.
4. Distributing the papers among those undertaking the refereeing.
5. Collecting the referees reports and selecting the papers for inclusion in the programme.
6. Grouping selected papers into sessions for presentation and selecting chairman for each session.

Organising Committee

1. Preparing a list of people to invite to the conference.

2. Issuing priority invitations to key attendees who are members of IFIP working groups..

Provide:

i. some scenarios [5 marks]

ii. a use case model [5 marks]

iii. use cases with any relevant extensions [5 marks]

iv. problem domain object list [5 marks]

v. a preliminary class diagram [5 marks]

vi. a sequence diagram for a selected use case [5 marks]

vii. state diagram for selected class [5 marks]

viii. final class diagram [5 marks]

ix. a selected revised use case [5 marks]

Q1 Part (b)

i. What is meant by traceability? [5 marks]

ii. Why do commercial software engineering organisations pay so much attention to the requirements phase of the "software life-cycle"? [5 marks]

iii. You are working on a project to develop software to assist in the management of secondary schools. The system will initially handle academic and financial records, but may well need to encompass other areas such as personnel , time tabling and stock control. In addition, you are aware that all areas of school management are entering a period of very considerable change. Your manager has in the past successfully developed other record-handling systems using the

waterfall model of development and is firmly convinced that it is again appropriate in this situation. What might be the arguments she would put forward to support her position and how might you argue for a different approach? [10 marks].

Q1 Part (c)

i. Distinguish between external and internal quality characteristics. Give examples to support your distinction. [5 marks]

ii. Outline the CMU-SEI process maturity framework. Describe the process improvement actions required in order to move from each level to the next. [5 marks]

iii. The double maintenance, shared data and simultaneous update problems are characteristic problems of configuration management. Briefly describe them. [5 marks]

iv. The National Air Traffic Services wish to procure a large software system for controlling en-route traffic across the Atlantic. Why might such a system be regarded as posing particular problem for the software developer? [5 marks]

CONTINUED

Question 2. Answer part (a) and *either* part (b) *or* part(c).[Total 35 marks]

Q2 Part (a)

i) Write down the set of values and *minimal* set of operations that define the abstract data types Queue and Stack.

[4 marks]

ii) Suppose we have a class `Queue` that implements a Queue abstract data type and contains values of Java type `Object`. What is the output of the following Java program:

```
class QueueTest {
    public static void main(String[] args) {
        Queue q = new Queue();
        q.enQueue('A');
        q.enQueue('B');
        q.enQueue('C');
        q.enQueue('D');
        while(!q.empty()) {
            System.out.println(q.deQueue());
        }
    }
}
```

[2 marks]

iii) The above program is adapted to use a `Stack` object, which implements the Stack abstract data type in a similar way, instead of a `Queue`. What is the output of the adapted program?

[2 marks]

Q2 Part (b)

The following table contains a set of key value pairs that are required to be stored in a look-up table:

Key	Value
Geoff	44
Simon	26
Mary	50
Adam	24
Barry	31
Wilma	48
Carla	39

i) The entries are to be stored in a binary search tree. Each entry is stored in an instance of a `TreeNode` class, which is implemented as follows:

```
class TreeNode {
    public TreeNode(String k, Object v) {
        key = k;
        value = v;
        left = right = null;
    }
    public void print() {
        System.out.println(key + ":" + value);
    }
    protected String key;
    protected Object value;
    protected TreeNode left;
    protected TreeNode right;
```

```
}
```

The binary search tree class itself consists of a single `TreeNode` object reference, `root`. Describe in words or pseudo code an appropriate method for inserting entries into the tree that enables them to be retrieved efficiently. Draw the data structure, which contains the tree after the entries in the table above have been added using your scheme. Indicate on your diagram where each individual entry is stored in the data structure.

[8 marks]

ii) The `TreeNode` class has a public interface method, `print`, which prints out the {key, value} pair it contains. Sketch out three simple recursive methods of the binary search tree class that print out all of the entries in the tree. Choose one method and write down the order in which the entries in the tree you have drawn for (i) are printed out when the method is called.

[6 marks]

iii) An alternative order in which the tree can be traversed is *level order*, which visits the nodes of the tree in order of their depth. Write down the keys of the binary search tree you drew in (i) in level order.

[2 marks]

iv) We can consider the tree as a special case of a digraph in which the nodes of the tree are the vertices of the graph and the parent to child links in the tree are the edges of the graph. Consider the tree you drew in (i) this way and perform (by hand) a breadth first traversal of the structure starting from the root. Write down the order in which the nodes are visited.

[2 marks]

v) Notice the connection between the answers to parts (iii) and (iv) and use this observation to outline an alternative implementation of the `print` method, which outputs the entries in level order. You may assume the existence of a class `Queue`, which implements the queue ADT.

[9 marks]

Q2 Part (c)

i) A digraph G = <V, E> is specified by the set V = {a, b, c, d, e} of vertices, and the set E = {(a, a), (b, a), (b, d), (c, b), (d, e), (e, c), (e, d)} of edges. Draw a diagram representing G, which clearly shows the direction of each edge, and list the set of simple, proper cycles of G.

[9 marks]

ii) Describe a data structure that could be used to represent a graph such as G and that is flexible enough to allow the addition of any number of new vertices and edges. In your description explain what Java classes you would use to represent the objects that comprise the structure, and how new vertices and edges are added to the graph. Draw the data structure that represents the graph G in (i) showing which parts of the structure correspond to which vertices and edges of G.

[11 marks]

iii) Describe how you could use a hash table to speed up operations on particular vertices of a graph stored in a data structure such as the one you drew for (ii). Explain why it helps.

[7 marks]