B4.3-R3: SOFTWARE TESTING AND QUALITY MANAGEMENT

NOTE:

1. Answer question 1 and any FOUR questions from 2 to 7.
2. Parts of the same question should be answered together and in the same sequence.

Time: 3 Hours **Total Marks: 100**

1. State whether the following statements are **TRUE** or **FALSE.** In each case, justify your answer using one or two sentences. Irrelevant and unnecessarily long answers will be penalized.

a) The terms software verification and software validation are essentially synonyms.

b) Introduction of additional sequence type of statements in a program can not increase its cyclomatic complexity.

c) Code walkthrough for a module is normally carried out after completion of unit test.

d) During code review you detect errors whereas during code testing you detect failures.

e) Branch coverage is a stronger testing technique compared to statement coverage technique.

f) Modern quality assurance paradigms are centered around to carryout thorough product testing.

g) A satisfactory way to test object-oriented programs is to test all the methods supported by the different classes individually and then by performing adequate integration and system testing.

(7x4)

2.

a) Explain why testing techniques used for traditional procedure-oriented programs can not effectively be used to test object-oriented programs? What additional types of tests are needed for object-oriented programs?

b) Explain the difference between code inspection and code walk through. Why is detection and correction of errors during inspection and walkthrough preferable to that achieved using testing?

c) Prepare a checklist that can be used for inspection of the user interface of a software product.

(6+6+6)

3.

a) What is the difference between the top-down and the bottom-up integration testing approaches? Explain your answer using an example. Why is the mixed integration testing approach preferred by many testers?

b) Design the black-box test suite for a program that accepts two strings and checks if the first string is a substring of the second string and displays the number of times the first string occurs in the second string.

c) Explain what do you understand by client-server software. What are its advantages over the traditional software architecture? How can a client-server software be effectively tested?

(6+6+6)

4. Consider the following program segement.

```
/* num is the number the function searches in a presorted integer array arr */
int bin_search(int num)
{
      int min,max;
      min =0;
      max =100;
      while(min!=max){
            if(arr[(min+max)/2]>num)
            max=(min+max)/2;
            else if(arr[(min+max)/2]<num)
            min=(min+max)/2;
                  else return((min+max)/2);
      }
      return(-1);
}
```

a) Draw the control flow graph for this program segment.

b) Determine the cyclomatic complexity for this program. (Show the intermediate steps in your computation. Writing only the final result is not sufficient)

c) How is the cyclomatic complexity metric useful?

(6+6+6)

5.

a) Explain, why measurement of software reliability is a much harder problem than the measurement of hardware reliability.

b) What do you understand by a reliability growth model? How is reliability growth modelling useful? Give examples of two reliability growth models.

c) Explain the importance of software configuration management in modern quality paradigms such as SEI CMM and ISO 9001. What problems might arise if a development organization does not use any configuration management tool?

(6+6+6)

6.

a) Explain two test coverage metrics for procedural code. How are these useful? Can these be used satisfactorily for object-oriented programs? Explain your answer.

b) Why effective testing of real-time and embedded systems is considered more difficult than testing traditional systems? Explain a satisfactory scheme to test real-time and embedded systems.

c) Distinguish between the static and dynamic analysis of a program. Explain at least one metric that a static analysis tool reports and one metric that a dynamic analysis tool reports. How are these metrics useful?

(6+6+6)

7.

a) What do you understand by volume testing? Explain using a suitable example how volume test cases can be designed and the types of defects these tests can help to detect.

b) Explain at least one defect metric and how this metric can be collected. Also explain how defects can be effectively tracked for a software product.

c) What do you understand by data flow testing? How is data flow testing performed? Is it possible to design data flow test cases manually? Explain your answer.

(6+6+6)