Nov- Dec - 2012

Total No. of Questions : 11]

SEAT No. :

P1206

[Total No. of Pages : 4

[4264] - 713

B.E. (I.T.)

SOFTWARE TESTING AND QUALITY ASSURANCE

(2008 Pattern) (Semester - I)

Time :3 Hours] *[Max. Marks :100*

Instructions to the candidates:

1) *Answer question number 1 or 2, 3 or 4, 5 or 6 from section - I.*
2) *Answer question number 7 or 8, 9 or 10, 11 from section - II.*
3) *Answers to the two sections should be written in separate answer books.*
4) *Neat diagrams must be drawn wherever necessary.*
5) *Figures to the right indicate full marks.*
6) *Assume suitable data, if necessary.*

SECTION - I

Q1) a) Explain in short any four methods of System Level Testing. **[8]**

b) Is complete testing possible? When to stop testing? Explain the difference between random testing and testing using error guessing. **[8]**

OR

Q2) a) Explain any four with example **[8]**

i) Error
ii) bug
iii) defect
iv) fault
v) failure
vi) test ware

b) 'V & V diagram is basis for every type of testing?' Comment on this statement. What is the role of test plans in a V & V diagram. **[8]**

P.T.O.

Q3) a) Consider the following program segment: **[8]**

```
Main( )
{
    Int number, index;
1.  Printf("Enter a number");
2.  scanf("%d";&number);
3.  index=2;
4.  while (index<=number-1)
5.  {
6.          if(number%index==0)
7.          {
8.              printf("Not a prime number")
9.              break;
10.         }
11.         index++;
12. }
13. if (index==number)
14.         printf("Prime number");
}//end main
```

i) Draw the Control Flow Graph for the program.

ii) Calculate the cyclomatic complexity of the program using all the methods.

iii) List all independent paths.

iv) Describe how cyclomatic complexity number and the flow graph be used to design a set of white box tests for this module.

b) Draw and explain defect life cycle. **[8]**

OR

Q4) a) Explain the importance of Boundary value analysis and Equivalence class partitioning with example. **[8]**

b) Draw and explain Software testing life cycle. **[8]**

Q5) a) Explain Product, Process and Resources with respect to their Internal and External attributes. **[9]**

b) Define measurement scale and explain the various scales of measurement. **[9]**

OR

Q6) Spell Check Specs: The checker accepts as input a document file and an optional personal dictionary file. The checker lists all words not contained in either of these files. The user can query the number of words processed and the number of spelling errors found at any stage during processing. **[18]**

	Weighting Factor		
Item	Simple	Average	Complex
External Inputs	3	4	6
External Outputs	4	5	7
External Inquiries	3	4	6
External Files	7	10	15
Logical Internal Files	5	7	10

There are 14 technical complexity factors out of that two factors has rating as 5 and six factors has rating as 3 and remaining six has rating as 0 on a scale of 0 to 5. Where 0 means irrelevant, 3 means it is average and 5 means it is essential to the system being built.

Based on the above perform the following:

a) Draw pictorial representation of the system for FP analysis.

b) Identify internal logical files, external I/P, O/P, Inquiries and Files.

c) Calculate Function Count (FC).

d) Calculate Technical Complexity Factor (TCF).

e) Calculate Function Point (FP).

f) Explain the importance of FP in software testing.

SECTION - II

Q7) a) Differentiate between quality control, quality assurance and quality management. **[8]**

b) Explain the objectives and elements of software reviews and inspections. **[8]**

OR

Q8) a) List and explain Ishikawa's Seven Basic Quality tools. **[8]**

b) List and explain any four Software reliability Quality Attributes. **[8]**

Q9) a) What is six sigma? Explain the terms DMAIC and DMADV with reference to six sigma. **[8]**

b) Explain the PDCA cycle in detail with reference to ISO 9000:9001. **[8]**

OR

*Q10)*a) List all the requirements of ISO 9000 and ISO 9001. **[8]**

b) What is SEI's Capability Maturity Model (CMM)? Explain each level in detail with their Key Process Area (KPA). **[8]**

*Q11)*Write short notes on any three: **[18]**

a) Software Configuration Management (SCM).

b) Quantitative Process management (QPM).

c) Software Product Engineering (SPE).

d) Software Project Tracking and Oversight (SPTO).

e) Process Change Management (PCM).