Reg. No.										

Manipal Institute of Technology, Manipal

(A Constituent Institute of Manipal University)

VII SEMESTER B.TECH (COMPUTER SCIENCE AND ENGINEERING)

MAKE UP SEMESTER EXAMINATIONS, DEC/JAN 2016

SUBJECT: SOFTWARE TESTING AND ANALYSIS (CSE 421)

REVISED CREDIT SYSTEM

Time: 3 Hours MAX. MARKS: 50

Instructions to Candidates:

- Answer **ANY FIVE FULL** questions.
- Missing data, if any, may be suitably assumed.

1.A. Consider a program for classification of a triangle. Its input is a triple of positive integers(say a,b,c) and the input parameters lie between 1 and 100. The triangle is classified according to the following rules: 3M
Right angled triangle: <sum of squares of two sides is equal to square of other>
obtuse angled triangle : <sum of squares of two sides is less than square of other>
acute angled triangle : <sum of squares of two sides is greater than square of other>
Output can be one of the following:
Right angled triangle, obtuse angled triangle, acute angled triangle, Invalid triangle.
Design efficient Boundary value test suite.

1.B. A program takes an angle within range [0,360] and determines in which quadrant the angle lies. Identify input domain based and output domain based equivalence classes. 3M

1.C. Explain adequacy criteria and test harness. Also briefly explain interclass testing with respect to OO testing. 4M

2.A. What is acceptance testing? Explain its types with an example. 4M

2.B. Explain in detail, redundancy principle with respect to software testing and analysis. 3M

2.C. Explain the significance of visibilty principle with respect to software testing and analysis. 3M

3.A. Explain the concept of lockset analysis 5M

3.B. For the given program in Fig 3B, 5M

a. Draw a CFG by considering each statement as a node[granularity is one].

b. Give all-c-uses criterion paths for ‘ti’ and all-p-uses criterion paths for ‘tv’ and generate the test cases the test cases for the same.

```
1.  public static double ReturnAverage(int value[], int AS, int MIN, int MAX) {
2.  int i, ti, tv, sum;
3.  double av;
4.  i = 0; ti = 0; tv = 0; sum = 0;
5.  while (ti < AS && value [i] != -999) {
6.  ti++;
7.  if (value[i] >= MIN && value[i} <= MAX) {
8.  tv++;
9.  sum = sum + value[i];
10. }
11. i++;
12. }
13. if (tv > 0)
14. av = (double) sum/tv;
15. else
16. av = (double) -999;
17. return (av)
18. }
```

Fig: 3B

4.A. Design separate test suites for Statement and Branch for program given in Fig 3B. 4M

4.B. Design test cases for Condition testing for program given in Fig 3B. 3M

4.C. Write the checklist for analyzing that the state machine is complete and consistent enough for model or implementation testing. 3M

5.A. Explain how inheritance and polymorphism is an issue in Object Oriented Testing. 4M

5.B. List typical control faults that need to be tested in state based testing. 2M

5.C A marketing company wishes to construct a decision table to decide how to treat clients according to three characteristics: Gender, City Dweller, and age group: A (under 30), B (between 30 and 60), C (over 60). The company has four products (W, X, Y and Z) to test market. Product W will appeal to female city dwellers. Product X will appeal to old females. Product Y will appeal to Male middle aged shoppers who do not live in cities. Product Z will appeal to all but young city dwellers. 4M

Design efficient decision table and also derive test cases based on the decision table.

6.A. Consider the given program P given in fig 6A 5M

```
1.  begin
2.      int x, z, i
3.      input (x,z);
4.      for(i=1; i<5; i++) {
5.      z=z+x*i
6.      }
7.     output( z );
8.  end
```

Fig 6A

P is assumed to be correct and hence the expected output is same as the actual output for P.

Test suite T is as follows

Test case Id	x	z	Expected
1	0	1	1
2	0	2	2

Rules for mutant generation are as follows

a. < is changed to <=
b. + is changed to – [**Note**: Do not consider ++]
c. * is changed to +

For the above given problem,

- Generate first order mutants and run the given test suite T for mutants generated.

(Clearly show the output of each intermediate step). Represent the results in tabular format wherever applicable.

6.B. For program in question **6.A,** find the mutation score and enhance the test suite, if required. 2M

6.C. What is test oracle? Explain the different types of test oracle with an example 3M

****