REG. NO										

MANIPAL INSTITUTE OF TECHNOLOGY
(Constituent Institute of Manipal University)
MANIPAL-576104

VII SEMESTER B.E.(COMPUTER SCIENCE AND ENGINEERING) DEGREE
END-SEMESTER EXAMINATION-DEC 13
SUBJECT: SOFTWARE TESTING AND ANALYSIS (CSE 421)
DATE: 06/12/2013

TIME: 3 HOURS MAX.MARKS: 50

Instructions to Candidates

- **Note:** Answer any **FIVE** full questions.

1.A. The Chipmunk marketing division is worried about the start-up time of the new version of the RodentOS operating system. The marketing division requirement suggests a software requirement stating that the start-up time shall not be annoying to the users. Another requirement is that, it has a calendar program that should provide timely reminders to the users. Explain why this simple requirement is not verifiable and try to reformulate the requirement to make it verifiable.

1.B. A riffle salesperson in the former Arizona territory sold riffle locks, stocks and barrels made by a gunsmith in Missouri. Locks cost $45, Stocks cost $30 and barrels cost $25. The sales person had to sell at least one complete riffle per month and production limits were such that the most a salesperson could sell in a month was 70 locks, 80 stocks and 90 barrels. After each town visit, the salesperson sent a telegram to the Missouri gunsmith with the number of locks, stocks and barrels sold in that town. The system uses locks=-1 to indicate end of input data. When the sales of the month were complete, the Gunsmith computes the salespersons commission as follows: 10% if sales are less than $1000, 15% if sales are between $1000 and $1800 and 20% if the sales are above $1800. The commission program produced a monthly sales report that gave the total number of locks, stocks and barrels sold, the salespersons total dollar sales and finally the commission.
For the above scenario,

a. Construct Normal Boundary Value Analysis (BVA) test cases.
b. Identify the valid and invalid classes of the input variables.
c. Write the set of weak normal and weak robust test cases.

(3+(2+2+3))

2.A. Explain the steps in constructing the decision table. For the scenario given in Q1B. , Construct a decision table and derive the test cases.

2.B Indicate which principle(s) of Software Testing guided the following choices. JUSTIFY.

a. Use an externally readable format also for internal files, when possible.
b. Collect and analyze data about faults revealed and removed from the code.
c. Produce complete fault reports.

(4+(2+2+2))

3.A. Explain with an example how the issue of inheritance and polymorphism is handled in OO testing.

3.B.
```
1.   static void questionable( ){
2.       int k,i,n;
3.       input (n,k);
4.       for(i=0; i<n; i++) {
5.           if( k<0))
6.               {k=0};
7.           else { k+=i}
         }
8.    system.out.println (k); }
```

For the above code,

a. Draw a Data Flow Graph.
b. Find DEF, C-USE and P-USE for every node in the graph and list all DU pairs.
c. Design test cases that cover *all-def, all-uses* and *all- DU pair* coverage criterion.

(2+(1+4+3))

4.A. For the given requirement R and program P, enhance the following test set T using the mutation operator M.

R: The program should print biggest of 3 numbers a,b,c. If the numbers are equal, it should print the first number in the sequence to be the biggest.

P:

```
void CompareThreeNos( )
{
 int a,b,c;
 input( a,b,c);
 if((a>=b) and (a<=c)) { print( a is biggest);}
 else if(b>=a) and (b>=c)  { print( b is biggest);}
 else{  print( c is biggest);}
}
```

T:

TC#	a	b	c	Exp O/P
1	0	0	0	a is biggest
2	3	1	2	a is biggest

M: Logical AND Operator is changed to OR**.** (Use First Order Mutants).

4.B. Explain the concept of Memory analysis and Lockset Analysis.

(5+5)

5.A. With an example, explain how partial oracles are different from comparison based oracles.

5.B. Derive test cases for Functionality M*aintenance* from the FSM specification in Fig 1.

a. Derive a test suite that satisfies the Transition Coverage criterion
b. Derive a test suite that satisfies the Single State Path Coverage criterion.
c. Indicate at least one element of the program that must be covered by a test suite satisfying the Single Transition Path Coverage, but need not be covered by a test suite that satisfies the Single State Path Coverage criterion. Derive a test case that covers that element

(3+(2+3+2))

6.A. Suppose test suite A satisfies adequacy criteria C1. Test suite B satisfies adequacy criteria C2, and C2 subsumes C1. Can we be certain that faults revealed by A will also be revealed by B? Explain.

6.B. For the code given in Q3.B, derive the test cases that satisfy boundary interior criterion and loop boundary adequacy criterion.

6.C. Explain i) Test execution ii)Adequacy Criterion

(3+5+2)

.

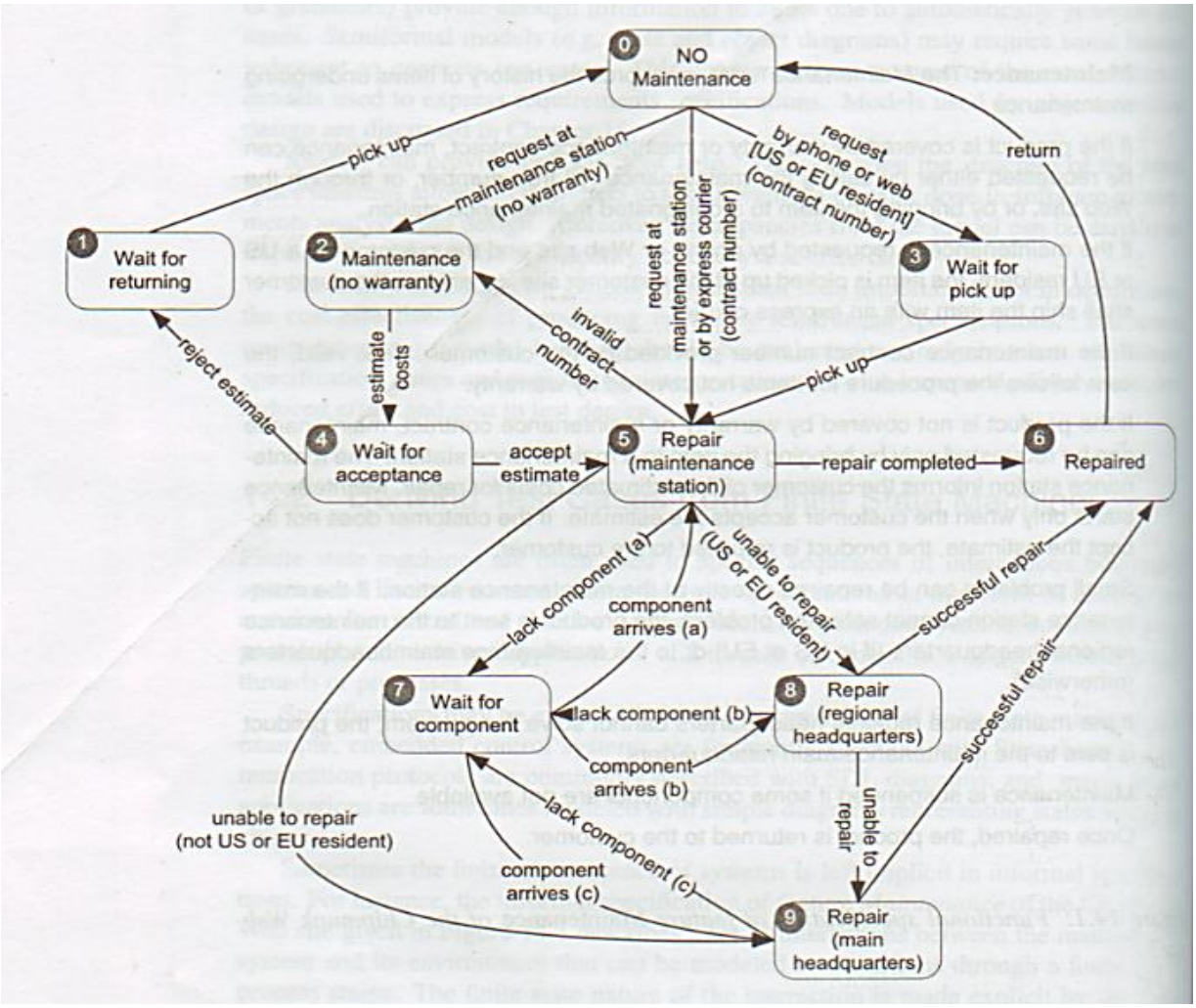

Fig 1.

****