Sr. No........................................

# YMCA UNIVERSITY OF SCIENCE& TECHNOLOGY, FARIDABAD

## M.Tech. CE & IT, I SEMESTER

## Software Testing (MTCE-17-105)

**Time: 3 Hours** **Max. Marks:75**

**Instructions:**
1. *It is compulsory to answer all the questions (1.5 marks each) of Part -A in short.*
2. *Answer any four questions from Part -B in detail.*
3. *Different sub-parts of a question are to be attempted adjacent to each other.*

## PART -A

Q1 (a) What are boundary related design bugs? (1.5)

(b) What do you mean by effective testing? (1.5)

(c) Which test plans are made corresponding to HLD and LLD? (1.5)

(d) What is the difference between function and system testing? (1.5)

(e) What is the significance of dc paths? (1.5)

(f) What is the role of stubs and drivers? (1.5)

(g) How do you calculate cyclomatic complexity? Give any there methods. (1.5)

(h) How do you calculate defect density if you know Function points in a project? (1.5)

(i) What is incremental testing in OO software? (1.5)

(j) What is APFD? (1.5)

## PART -B

Q2 (a) 'V & V diagram is basis for every type of testing'. Comment and explain in detail on this statement. (10)

(b) Demonstrate that invalid inputs and unexpected behavior have a high probability of finding an error. (5)

Q3 (a) A program has been designed to determine the nature of roots of a quadratic equation. It takes three input values from the range [0,100].
Design all test cases for this program using BVC, Robust testing and worst-case testing methods. (3+3 +4)

b) What is the significance of post implementation phase in STLC? (5)

Q4 Consider the following program for multiplication of two matrices:

```
main()
{
        int a[SIZE] [SIZE], b[SIZE] [SIZE], c[SIZE [SIZE], i, j, k, rowl, colml, row2, colm2;

        printf("Enter the order of first matrix <= %d %d \n", SIZE, SIZE);
        scanf("%d%d",&row1, colm1);
        printf("Enter the order of second matrix <= %d %d \n", SIZE, SIZE);
        scanf("%d%d",&row2, colm2);
        if(colm1==row2)
        {
                printf("Enter first matrix");
                for(i=0; i<row1;i++)
                {
                          for(j=0; j<colml; j++)
                                    scanf("%d", &a[i][j]);
                }
                printf("Enter second matrix");
                For(i=0; i<row2;i++)
                {
                          for(j=0; j<colm2; j++)
                                    scanf("%d", &b[i][j]);
                }
                printf("Multiplication of two matrices is");
                for(i=0; i<row1; i++)
                {
                          for(j=0; j<colm1; j++)
                          {
                                    c[i] [j] = 0;
                                    for(k=0; k<row2; k++)
                                              c[i][j] += a[i][k] + b[k][j];
                                    printf("%6d", c[i][j]);
                          }
                 }
        }
        else
        {
                 printf(" Matrix multiplication is not possible");
        }
}
```

a) Draw the DD graph for the program. (5)

b) Calculate the cyclomatic complexity of the program using any two methods. (3)

c) List all independent paths. (2)

d) Design all test cases from independent paths. (2)

e) Derive all du-paths and dc-paths using data flow testing. (3)

Q5 (a) Explain N-fold inspection? What are the factors that increase its effectiveness? (5)

(b) In a project, the estimated function points are 1200. Calculate the total number of test cases in the system and number of test cases in acceptance testing. Also calculate the defect density (number of total defects is 236) and test case coverage (10)

Q6 (a) Explain in detail coverage based test case prioritization method. (10)

(b) What are the guidelines for automated testing? (5)

Q7 (a) Explain the issues in testing inheritance features in an OO software. (10)

(b) Explain various debugging techniques? (5)

**************