



VIT[®]
Vellore Institute of Technology
(Approved to be University under section 3 of the UGC Act, 1956)

Final Assessment Test – May 2024

Course: **CSI3014 - Software Verification and Validation**

Class NBR(s): **2425/2509**

Slot: **B1+TB1**

Time: **Three Hours**

Max. Marks: **100**

- KEEPING MOBILE PHONE/ELECTRONIC DEVICES EVEN IN 'OFF' POSITION IS TREATED AS EXAM MALPRACTICE
- DON'T WRITE ANYTHING ON THE QUESTION PAPER

Answer any **TEN** Questions

(10 X 10 = 100 Marks)

1. Explain a generic process framework that supports to develop any software the foundation for a project in detail.
2. Justify the importance of requirements validation in the software engineering process. Illustrate its types and techniques in detail.
3. Convert the given set of requirements from natural language specification to formal language specification.
A university is admitting students in a professional course subject to the following conditions:
 - (R1) Marks in Java ≥ 70
 - (R2) Marks in C++ ≥ 60
 - (R3) Marks in OOAD ≥ 60
 - (R4) Total in all three subjects ≥ 220 OR Total in Java and C++ ≥ 150
 If the aggregate mark of an eligible candidate is more than 240, he will be eligible for scholarship course, otherwise he will be eligible for normal course.
4. Show how the defect management process can expose the defects at an early stage of the software development process, so as to reduce the effects on the software.
5. State and elaborate the need for automatic static analysis. Show how available expression, live variable and taint analysis are analyzed by the automated static analysis tool. Support your answer with suitable examples.
6. A program takes an angle as input within the range [0, 360] and determines in which quadrant the angle lies. Design test cases using equivalence class partitioning method.
7. Derive test cases by applying Basis Path Testing for the code given below.

```
#include <stdio.h>
main()
{
    1 int a[SIZE][SIZE], b[SIZE][SIZE], c[SIZE][SIZE], i, j, k, row1, colm1, row2, colm2;
    2 printf("Enter the order of first matrix <= %d %d \n", SIZE, SIZE);
    3 scanf("%d%d",&row1, colm1);
    4 printf("Enter the order of second matrix <= %d %d \n", SIZE, SIZE);
    5 scanf("%d%d",&row2, colm2);
    6 if(colm1==row2)
    7 {
    8 printf("Enter first matrix");
    9 for(i=0; i<row1; i++)
```

```

10 {
11 for(j=0; j<colm1; j++)
12 scanf("%d", &a[i][j]);
13 }
14 printf("Enter second matrix");
15 for(i=0; i<row2; i++)
16 {
17 for(j=0; j<colm2; j++)
18 scanf("%d", &b[i][j]);
19 }
20 printf("Multiplication of two matrices is");
21 for(i=0; i<row1; i++)
22 {
23 for(j=0; j<colm1; j++)
24 {
25 c[i][j] = 0;
26 for(k=0; k<row2; k++)
27 c[i][j] += a[i][k] + b[k][j];
28 printf("%6d", c[i][j]);
29 }
30 }
31 }
32 else
33 {
34 printf("Matrix multiplication is not possible");
35 }
36 }

```

8. a) "SAAS companies use Iterative testing as a tool to strengthen their product" – Justify the statement with suitable examples. [5]
b) "Defect Seeding is considered as a reliability measurement for the release of a software product" - Justify the statement with suitable examples. [5]
9. Illustrate how does the quality of software development process directly affects the quality of the delivered product. Discuss the major categories of cost of quality and explain the benefits of investment on quality.
10. "Test automation can solve much of the tester's challenges" – Justify the statement with neat illustration. Design an architecture for test automation to justify the statement.
11. "Can software inspection replace software testing" – Discuss the statement with neat examples. Illustrate the use of stubs and drivers in the decomposition-based integration testing with a good example.
12. Justify the need for metrics in a software testing process and discuss the metrics used to assess the project's overall quality.

⇔⇔⇔H/E/TX⇔⇔⇔