B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

April 2025 Semester End Make-Up Examinations

Programme: B.E.
Branch: Information science and engineering
Course Code: 23IS5PCSTG
Course: Software testing
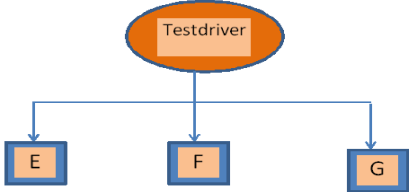
Semester: V
Duration: 3 hrs.
Max Marks: 100

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

			UNIT - I	CO	PO	Marks
1	a)		Imagine you're preparing to launch a new software application. What are the various **testing activities** involved in ensuring it's ready for release?	CO1		5
	b)		Suppose you're developing a new e-commerce website. Identify and explain the **different levels of testing** involved to ensure the website's functionality, performance, and security before its launch, with a suitable diagram.	CO1		5
	c)		Using a standard test case template, identify three test scenarios for an online e-commerce application i.e, Amazon and develop four functional test cases for each scenario	CO2	PO1	10
			OR			
2	a)		Justify how different **perspectives on software quality** can be applied when choosing a new laptop for personal or professional use.	CO1		5
	b)		Differentiate between Verification and Validation.	CO1		5
	c)		Using a standard test case template, identify three test scenarios for a mobile banking application and design four functional test cases for each scenario.	CO4	PO3	10
			UNIT - II			
3	a)		Differentiate between Static and dynamic testing.	CO1		5
	b)		i) Write the java code to test the unit which adds two numbers successfully or not in JUnit framework.	CO2	PO1	5

		ii) Write JUnit test cases to verify the correctness of the addition for different scenarios (positive numbers and negative numbers).			
	c)	Develop three mutants for a program provided below that finds prime numbers or not, and find if the mutants are killable, stubborn or equivalent mutant for 3 inputs each. Calculate the final mutation score. ```c\n#include<stdio.h>\n\nbool is_prime(int n) {\n if (n <= 1) {\n return false;\n }\n for (int i = 2; i * i <= n; i++) {\n if (n % i == 0) {\n return false;\n }\n }\n return true;\n}\n```	CO3	PO2	**10**

<div align="center">OR</div>

4	a)	Differentiate between Inspection and walkthrough.	CO1		**5**
	b)	i) Write the java code to test the unit which subtracts two numbers successfully or not in JUnit framework. ii)Write JUnit test cases to verify the correctness of the subtraction for different scenarios (positive numbers and negative numbers).	CO2	PO1	**5**
	c)	Explain the different steps involved in the code review process in static testing with a diagram.	CO1		**10**

<div align="center">UNIT - III</div>

5	a)	Illustrate **Robust Worst case boundary value Testing** for a function of two variables with a diagram.	CO1		**4**
	b)	Design robust boundary value analysis test cases for an online learning platform that accepts registrations from individuals aged 16 to 50 years and with heights ranging from 130 to 200 cm.	CO4	PO3	**6**
	c)	Illustrate **Weak Normal Equivalence Class Testing** with diagram.	CO1		**4**
	d)	Design test cases using strong robust ECT for triangle problem with sides ranging from (1-999 cms) considering both lower limit and upper limit values.	CO4	PO3	**6**

<div align="center">OR</div>

6	a)	Design normal boundary value analysis test cases for a registration page of a fitness club that accepts individuals aged between 18 and	CO4	PO3	**6**

		45 years , weight ranging from 40-100 kgs and with heights ranging from 150 to 180 cm.			
	b)	Design worst-case boundary value analysis test cases (15 test cases) for an airline booking system that allows passengers aged between 12 and 65 years, weighing between 20-150 kg, and with baggage sizes ranging from 50 to 200 cm in total dimensions	CO4	PO3	**6**
	c)	Illustrate **Strong Normal Equivalence Class Testing** with diagram.	CO1		**4**
	d)	Design test cases using weak robust ECT for triangle problem with sides ranging from (1-100 cms) considering both lower limit and upper limit values.	CO4	PO3	**4**
		UNIT - IV			
7	a)	What is Katalon studio? Why do we use katalon?	CO1		**4**
	b)	Illustrate with an example the concept of Sandwich and Big Bang testing.	CO1		**6**
	c)	Explain any 5 different types of Interface errors.	CO1		**10**
		OR			
8	a)	Identify and list the key features of jmeter.	CO1		**4**
	b)	System Integration testing is performed at different levels of granularity. Justify with an example for each level of granularity	CO2	PO1	**6**
	c)	Apply bottom-up and top-down integration for the module hierarchy for the given figure. Compare both the approaches with reference to various criteria.	CO2	PO1	**10**
		UNIT - V			
9	a)	List and explain the McCall's quality factor and criteria.	CO1		**10**
	b)	You are a QA lead at a software company working on a retail management system for a client. The project is nearing completion, and the client has requested a User Acceptance Testing (UAT) .Explain how the **UAT process** is conducted to ensure the system meets their business needs.	CO3	PO2	**10**
		OR			
10	a)	List and explain the ISO 9126 Quality Characteristics.	CO1		**6**

		b)	Imagine you are a Test Manager for a logistics company developing a new shipment tracking system. The development phase is complete, and the project is now in the test execution phase. Describe how you would manage the test execution process for this project.	*CO3*	*PO2*	**4**
		c)	Explain how you would create a UAT test plan for an e-commerce platform to ensure the platform meets the client's business needs.	*CO1*		**10**

******