

(A constituent unit of MAHE, Manipal)

VII SEMESTER B. TECH (COMPUTER SCIENECE AND ENGINEERING) MAKEUP EXAMINATION, 22-FEB-2022 (PART B)

SUBJECT: SOFTWARE TESTING AND ANALYSIS (CSE 4076)

REVISED CREDIT SYSTEM

Time: 85 Minutes (75 (Write up) +10 (Upload)) (2.20 PM-3.45 PM)

Note: Answer ALL the questions.

- Students have to write the answers in their own handwriting on a white sheet of paper. Scanned handwritten answer paper has to be uploaded as single pdf file. Answer sheet should not be mailed or uploaded in chat.
- The following details should be mandatorily written in the answer paper: Name, Semester, Section, Roll No., Registered No., Subject name, and Signature with date.

1A	Consider a problem for the determination of the nature of roots of a quadratic equation. Its input is	
	a triple of positive integers (say a, b and c) and values may be from interval [0, 100]. The output	
	may have one of the following words: Not a quadratic equation, Real roots, Imaginary roots, Equal	
	roots. (Hint: Determinant (D) =b ² -4ac. If D=0 roots are equal, if D>0 roots are real.)	4M
	Design boundary value analysis test cases and robust test cases.	41111
1B	Create efficient set of equivalence classes and generate strong normal test cases for the problem in	21/1
	Q1A.	21 VI
1C	Create an efficient limited entry decision table for the problem given in Q1A. And also derive	4M
	the test cases out of it.	TIVI
2A	With the help of any example show how is DU path testing different when compared to DU pair	2М
	testing.	21 VI
2B	<u>Requirement R</u> : The program takes two variables x and y as input and checks if the value of x is	
	less than y+1. If TRUE, it returns the square of x as output. If FALSE, it displays 2*y as the result.	
	Program P:	
	1. #include <stdio.h></stdio.h>	
	2. int main()	
	3. {	
	4. int x, y;	
	5. printf("Enter the values of x and y:");	
	6. $scanf("%d%d",&x,&y);$	
	7. $if(x < y+1)$	
	8. printf("The square is %d",x*x);	
	9. else	
	10. printf("The result is %d",2*y);	
	11. return 0;	
	12. }	
	<u>Test Set T</u> = {(t1:x=1, y=0) (t2:x=2, y=2) (t3: x=1, y=3)}	
	<u>Mutants</u>	
	M1 : replace x with $x+1$ in line 7	
	M2 : replace $*$ with $+$ in line 8	5M
	M3 : replace \leq with \leq in line 7	5111

MAX. MARKS: 20

	For the above data, test the adequacy of the Test Set 'T' using mutation Testing. Enhance the test	
20	cases il required.	
2C	Consider the program modules shown below	
	main()	
	int a, b, c, sum, diff, mul;	
	scanf("%d %d %d", &a, &b, &c);	
	sum=findsum(a, b, c);	
	diff=minus(a, b, c);	
	mul=findprod(a, b, c); }	
	printf("%d %d %d", sum, diff, mul);	
	$\left\{ \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \right\}$	
	findsum (int x, int y, int z)	
	int d;	
	return (d);	
	}	
	(i) If main () module is not ready for the testing of findsum () module, how is the testing	
	performed?	
	(ii) Design stubs when modules minus () and findprod () are not ready when called in	274
	main().	3M