MUMBAI EDUCATIONAL TRUST

MET Institute of Computer Science

M.C.A Semester – III

Roll No :	1266
Name of Student :	Aman Verma
Subject :	MCAL35 Software Testing & Quality Assurance Lab
Module:	1
Date of Assignment:	26-08-2022
Deadline of Submission:	16-09-2022

Q. 1 A program determines the next date in the calendar. Its input is entered in the form of <mm/dd/yyyy> with the following range:

1<= mm<=12

1<=dd<=31

1900<=yyyy<=2025

Its output would be the next date or it will display 'invalid date'. Design test cases for this program using BVC, robust testing.

OUTPUT:

Number of variable=3

BVC:

	month	date	year
Min value	1	1	1900

Min+1 value	2	2	1901
Max value	12	31	2025
Max-1 value	11	30	2024
Nominal value	6	16	2010

Test cases using BVC 4n+1 =4(3)+1 =13

Test case id	Month	Date	Year	Expected Output
1	1	16	2010	01/17/2010
2	2	16	2010	02/17/2010
3	12	16	2010	12/17/2010
4	11	16	2010	11/17/2010
5	6	1	2010	06/02/2010
6	6	2	2010	06/03/2010
7	6	31	2010	01/07/2010
8	6	30	2010	01/07/2010
9	6	16	1900	06/17/1900
10	6	16	1901	06/17/1901
11	6	16	2025	06/17/2020
12	6	16	2024	06/17/2024
13	6	16	2010	06/17/2010

Robust Testing

	Month	date	year
Min value	1	1	1900
Min+1 value	2	2	1901
Min-1 value	0	0	1899
Max value	12	31	2025
Max+1 value	13	32	2026
Max-1 value	11	30	2024
Nominal value	6	16	2010

Test cases using Robust Testing 6n+1 =6(3)+1 =19

Test case id	Month	Date	Year	Expected Output
1	1	16	2010	01/17/2010
2	2	16	2010	02/17/2010
3	0	16	2010	Month is invalid
4	12	16	2010	12/17/2010
5	13	16	2010	Month is invalid
6	11	16	2010	11/17/2010
7	6	1	2010	06/02/2010
8	6	2	2010	06/03/2010
9	6	0	2010	Date is invalid
10	6	31	2010	07/01/2010

11	6	32	2010	Date is invalid
12	6	30	2010	07/01/2010
13	6	16	1900	06/17/1900
14	6	16	1901	06/17/1901
15	6	16	1899	Year is invalid
16	6	16	2025	06/17/2025
17	6	16	2026	Year is Invalid
18	6	16	2024	06/17/2024
19	6	16	2010	06/17/2010

Q.2 A program determines roots of quadratic equations, its input is three integers a, b, and c. (Quadratic equation $ax^2 + bx + c$) The values of variables are in interval [1,100]. The program output may have one of the following:

- Not a quadratic equation
- Real roots
- Equal roots
- Imaginary roots

Design test cases for this program using BVC, robust testing.

BVC

	a	b	c
Min value	1	1	1
Min+1 value	2	2	2
Max-1 value	99	99	99
Max value	100	100	100

Nominal value	48	51	49

Test case id	a	b	c	Expected Output
1	48	51	1	Distinct Roots
2	48	51	2	Distinct Roots
3	48	51	99	Imaginary Roots
4	48	51	100	Imaginary Roots
5	1	51	49	Distinct Roots
6	2	51	49	Distinct Roots
7	99	51	49	Imaginary Roots
8	100	51	49	Imaginary Roots
9	48	1	49	Imaginary Roots
10	48	2	49	Imaginary Roots
11	48	99	49	Distinct Roots
12	48	100	49	Distinct Roots
13	48	51	49	Imaginary Roots

Robust Testing

	a	b	c
Min-1 value	0	0	0
Min value	1	1	1
Min+1 value	2	2	2
Max-1 value	99	99	99
Max value	100	100	100
Max+1 value	101	101	101
Nominal value	48	49	51

Output :

Test case id	a	b	c	Expected Output
1	48	51	0	Invalid Input
2	48	51	1	Distinct Roots
3	48	51	2	Distinct Roots
4	48	51	99	Imaginary Roots
5	48	51	100	Imaginary Roots
6	48	51	101	Invalid Input
7	0	51	49	Invalid Input
8	1	51	49	Distinct Roots
9	2	51	49	Distinct Roots
10	99	51	49	Imaginary Roots
11	100	51	49	Imaginary Roots

12	101	51	49	Invalid Input
13	48	0	49	Invalid Input
14	48	1	49	Imaginary Roots
15	48	2	49	Imaginary Roots
16	48	99	49	Distinct Roots
17	48	100	49	Distinct Roots
18	48	101	49	Invalid Input
19	48	51	49	Imaginary Roots

Q.3 A mobile phone service provider uses a program that computes the monthly bill of customers as follows:
Minimum Rs 300 for up to 120 calls
Plus Rs.1 per call for next 70 calls
Plus Rs 0.80 per call for the next 50 calls
Plus Rs.0.40 per call for calls beyond 240.
Design test cases for this program using equivalence class testing technique.

The partition for the domain input as valid and invalid values are :

C1= No. of calls <=120 then Rs. 300

C2= No. of calls > 120 then for next 70 calls -> 300+(n-120)*1

C3= No. of calls >190 then for next 50 calls -> 300+70*1 + (n-190)*0.80

C4 = No. of calls > 240 then 300+70*1 + 50* 0.80 + (n-200)* 0.40

C5= No. of calls < 0 , invalid input

Test Case ID	No. of calls	Expected Results	Classes covered by the Test Case
1	70	Rs. 300	C1
2	150	Rs. 330	C2
3	210	Rs. 386	C3
4	250	Rs. 430	C4
5	-1	Invalid input	C5

Q.4 A program accepts three integer numbers a, b, and c as sides of a triangle. Check whether given value for an Equilateral, Isosceles, and Scalene triangle or can't form a triangle. Design test cases for this program using equivalence class testing technique.

The partition for the domain input as valid and invalid values are :

I1={<a,b,c>|a+b>c & b+c>a & a+c>b}

I2={<a,b,c>|a=b=c}

I3={<a,b,c>|a=b,a!=c}

I4={<a,b,c>|b=c,b!=a}

I5={<a,b,c>|a=c,a!=b}

I6={<a,b,c>|a!=b!=c}

I7={<a,b,c>|a+b<c}

I8={<a,b,c>| b+c<a}

I9={<a,b,c>| a+c<b}

Test case id	a	b	c	Expected output	Classes Tested (I)
1	5	4	3	Triangle,Scalene	I1,I6
2	4	4	3	Triangle,Isosceles	I1,I3
3	5	5	5	Triangle,Equilateral	I1,I2
4	3	4	4	Triangle,Isoscel	I1,I4

				es	
5	4	3	4	Triangle,Isoscel es	I1,I5
6	1	2	4	Not a triangle	I7
7	4	1	2	Not a triangle	I8
8	1	4	2	Not a triangle	I9