IT-314

Software Engineering

Lab 9: Equivalence Class Testing and Boundary Value Analysis

Vraj Gandhi (202201425)

Question 1

Equivalence Class Test Cases:

Test Case No	Day	Month	Year	Expected Output	Equivalence Classes
1	15	5	2000	14-05-2000	Valid date
2	1	1	2000	31-12-1999	Boundary test, new year
3	29	2	2004	28-02-2004	Leap year February
4	1	3	2004	29-02-2004	Leap year, prev date is 29-02
5	31	12	2015	30-12-2015	End of year
6	31	4	2000	Invalid Date	Invalid day for April (only 30 days)
7	31	2	2000	Invalid Date	February cannot have 31 days
8	29	2	2001	Invalid Date	Non-leap year February
9	0	5	2000	Invalid Date	Invalid day, less than 1
10	15	13	2000	Invalid Date	Invalid month, greater than 12
11	15	5	1899	Invalid Date	Invalid year, less than 1900
12	15	5	2016	Invalid Date	Invalid year, greater than 2015

Question 2:

Program-1: Linear Search

Test Case No	Input value	Array	Expected Output
1	5	{}	-1
2	3	{3}	0
3	7	{1,2,3,4,5,6}	-1
4	4	{1,2,3,4,5,6}	3
5	2	{2,2,2,2}	0
6	9	{5,7,9,12,9,15}	2
7	4	{10,11,12}	-1
8	12	{12}	0

Program-2: Counting

Test Case No	Input value	Array	Expected Output
1	5	{}	0
2	3	{3}	1
3	7	{1,2,3,4,5,6}	0
4	4	{1,2,3,4,4,4,6}	3
5	2	{2,2,2,2}	4
6	9	{5,7,9,12,9,15}	2
7	4	{10,11,12}	0
8	12	{12}	1

Program-3: Binary Search

- **Equivalence Classes:**
  - **A1**: The array is empty.
  - **A2**: The array has one element, and v is present.
  - **A3**: The array has one element, and v is not present.
  - **A4**: The array has multiple elements, and v is present in the middle.
  - **A5**: The array has multiple elements, and v is present at the beginning.
  - **A6**: The array has multiple elements, and v is present at the end.
  - **A7**: The array has multiple elements, and v is not present.
  - **A8**: The array has duplicate elements, and v is present.
  - **A9**: The array has duplicate elements, and v is not present.

Test Case No	Array (a[])	v (Search Value)	Expected Output	Covered Classes
1	[]	5	-1	A1
2	[7]	7	0	A2
3	[7]	5	-1	A3
4	[1, 3, 5, 7, 9]	5	2	A4
5	[1, 3, 5, 7, 9]	1	0	A5
6	[1, 3, 5, 7, 9]	9	4	A6
7	[1, 3, 5, 7, 9]	8	-1	A7
8	[1, 3, 3, 3, 9]	3	2	A8
9	[1, 3, 3, 3, 9]	4	-1	A9

Program-4: Triangle Classification

- **Input: Triangle validity**
  - **Class T1**: Invalid triangle (any side is greater than or equal to the sum of the other two sides).
  - **Class T2**: Valid triangle.
- **Input: Triangle type**
  - **Class E1**: Equilateral triangle (all sides are equal).
  - **Class E2**: Isosceles triangle (exactly two sides are equal).
  - **Class E3**: Scalene triangle (all three sides are different).
- **Input: Side lengths**
  - **Class S1**: Positive integers for all sides.

Program-5: Substring prefix

- **Input: Length of s1 and s2**
  - **Class L1**: s1 is longer than s2 (impossible to be a prefix).
  - **Class L2**: s1 is equal in length to s2 (potentially a prefix).
  - **Class L3**: s1 is shorter than s2 (can be a prefix).
- **Input: String contents**
  - **Class C1**: s1 is empty (prefix of any string).
  - **Class C2**: s2 is empty (only true if s1 is also empty).
  - **Class C3**: Both strings are non-empty.
  - **Class C4**: s1 is completely equal to s2 (valid prefix).

Test Case No	s1 (Value)	s1 Length	s2 (Value)	s2 Length	Expected Output	Covered Classes
1	""	0	"hello"	5	True (Empty s1 is a prefix of any s2)	L3, C1
2	"abc"	3	"abc"	3	True (s1 is exactly equal to s2)	L2, C4
3	"abc"	3	"abcdef"	6	True (s1 is a prefix of s2)	L3, C3
4	"abc"	3	"abxdef"	6	False (s1 is not a prefix of s2)	L3, C3
5	"abcdef"	6	"abc"	3	False (s1 cannot be a prefix of s2)	L1, C3
6	""	0	""	0	True (Both s1 and s2 are empty)	L2, C2
7	"abc"	3	"abcd"	4	False (s1 is not equal to s2)	L3, C3
8	"abc"	3	"abcde"	5	True (s1 is a valid prefix of s2)	L3, C3
9	""	0	"a"	1	False (Non-empty s2, empty s1)	L3, C1
10	"abcdef"	6	"abc"	3	False (s1 is longer, impossible to be a prefix)	L1, C3

Program-6: Triangle Classification

a) Identify the Equivalence Classes for the System:

1. **Valid Triangle Types:**
   - **E1**: Equilateral triangle (A = B = C)
   - **E2**: Isosceles triangle (A = B or A = C or B = C)
   - **E3**: Scalene triangle (A ≠ B ≠ C)
   - **E4**: Right-angled triangle ($A^2 + B^2 = C^2$ or any permutation)
2. **Invalid Cases:**
   - **I1**: Non-triangle (A + B ≤ C or A + C ≤ B or B + C ≤ A)
   - **I2**: Non-positive lengths (A ≤ 0, B ≤ 0, or C ≤ 0)

b) Identify Test Cases to Cover the Identified Equivalence Classes:

Test Case No	Side A	Side B	Side C	Expected Output	Notes
1	3	3	3	Equilateral	E1
2	5	5	3	Isosceles	E2
3	4	5	6	Scalene	E3
4	5	12	13	Right-angled	E4
5	2	2	5	Non-triangle	I1
6	1	1	3	Non-triangle	I1
7	0	5	5	Non-positive	I2
8	-1	2	2	Non-positive	I2
9	4	4	4	Equilateral	E1
10	0	0	0	Non-positive	I2

c) Boundary Condition for A + B > C (Scalene Triangle):

Test Case No	Side A	Side B	Side C	Expected Output	Notes
11	2	3	4	Scalene	A + B > C
12	3	4	5	Scalene	A + B > C
13	1	2	2	Isosceles	A + B = C
14	3	5	7	Scalene	A + B > C

d) Boundary Condition for A = C (Isosceles Triangle):

Test Case No	Side A	Side B	Side C	Expected Output	Notes
15	5	3	5	Isosceles	A = C
16	1	1	2	Non-triangle	A + B = C
17	4	4	2	Isosceles	A = B

e) Boundary Condition for A = B = C (Equilateral Triangle):

Test Case No	Side A	Side B	Side C	Expected Output	Notes
18	2	2	2	Equilateral	A = B = C
19	0	0	0	Non-positive	All sides zero

f) Boundary Condition for $A^2 + B^2 = C^2$ (Right-angled Triangle):

Test Case No	Side A	Side B	Side C	Expected Output	Notes
20	3	4	5	Right-angled	$A^2 + B^2 = C^2$
21	6	8	10	Right-angled	$A^2 + B^2 = C^2$
22	5	12	13	Right-angled	$A^2 + B^2 = C^2$

g) Test Cases for Non-triangle Case:

Test Case No	Side A	Side B	Side C	Expected Output	Notes
23	1	1	3	Non-triangle	A + B = C
24	5	10	20	Non-triangle	A + B < C

h) Test Cases for Non-positive Input:

Test Case No	Side A	Side B	Side C	Expected Output	Notes
25	-1	5	5	Non-positive	Negative length
26	2	-3	2	Non-positive	Negative length
27	0	4	4	Non-positive	Zero length