IT314

Q.1. Equivalence Class Test Cases for the Previous Date Program:

Equivalence Partitioning:

1. Valid Input:

- (1, 1, 1900) -> Previous Date: (31, 12, 1899)
- (1, 1, 2015) -> Previous Date: (31, 12, 2014)   -

(1, 3, 2000) -> Previous Date: (29, 2, 2000)   -

(15, 8, 2023) -> Previous Date: (14, 8, 2023)

2. Invalid Input:

- (0, 1, 2000) -> Invalid Date
- (1, 0, 2000) -> Invalid Date
- (1, 13, 2000) -> Invalid Date   - (32, 3, 2000) -> Invalid Date

Boundary Value Analysis:

1. Boundary Values:

- (1, 1, 1900) -> Previous Date: (31, 12, 1899)
- (1, 1, 2015) -> Previous Date: (31, 12, 2014)
- (1, 12, 1900) -> Previous Date: (30, 11, 1900)
- (31, 12, 1900) -> Previous Date: (30, 11, 1900)
- (1, 3, 1900) -> Previous Date: (28, 2, 1900)
- (29, 2, 2000) -> Previous Date: (28, 2, 2000)

- (30, 2, 2000) -> Invalid Date

- (31, 4, 2000) -> Previous Date: (30, 4, 2000)

- (31, 1, 2000) -> Previous Date: (30, 12, 1999)

Q.2. Test Cases for the Given Programs:

P1. Linear Search:

1. Equivalence Partitioning:

   - Element found in the array: [1, 2, 3, 4, 5], target = 3 -> Output: 2   - Element not found in the array: [1, 2, 3, 4, 5], target = 6 -> Output:---2

2. Boundary Value Analysis: - Empty array: [] -> Output: ----2

   - Array with single element: [5], target = 5 -> Output: 0 ----3

   - Array with single element: [5], target = 6 -> Output: -------3

P2. Count Item:

1. Equivalence Partitioning:

   - Element found in the array: [1, 2, 3, 2, 1], target = 2 -> Output: 2

   - Element not found in the array: [1, 2, 3, 4, 5], target = 6 -> Output: 0

2. Boundary Value Analysis:

   - Empty array: [] -> Output: 0

-

- Array with single element: [5], target = 5 -

> Output: 1

- Array with single element: [5], target = 6 -

> Output: 0

P3. Binary Search:

1. Equivalence Partitioning:

- Element found in the sorted array: [1, 2, 3, 4, 5], target = 3 -> Output: 2
- Element not found in the sorted array: [1, 2, 3, 4, 5], target = 6 -> Output: -1

2. Boundary Value Analysis:

- Empty array: [] -> Output: -1
- Array with single element: [5], target = 5 -> Output: 0
- Array with single element: [5], target = 6 -> Output: -1
- Sorted array: [1, 2, 3, 4, 5], target = 1 -> Output: 0 - Sorted array: [1, 2, 3, 4, 5], target = 5 -> Output: 4

P4. Triangle Classification:

1. Equivalence Partitioning:

- Equilateral Triangle: (5, 5, 5) -> Output: EQUILATERAL

  Isosceles Triangle: (5, 5, 7) -> Output: ISOSCELES
- Scalene Triangle: (3, 4, 5) -> Output: SCALENE
- Invalid Triangle: (1, 2, 3) -> Output: INVALID

2. Boundary Value Analysis:

- Equilateral Triangle: (1, 1, 1) -> Output: EQUILATERAL

- Isosceles Triangle: (1, 1, 2) -> Output: ISOSCELES

- Isosceles Triangle: (1, 2, 1) -> Output: ISOSCELES

- Isosceles Triangle: (2, 1, 1) -> Output: ISOSCELES

- Scalene Triangle: (3, 4, 7) -> Output: SCALENE

- Invalid Triangle: (1, 2, 4) -> Output: INVALID

- Invalid Triangle: (2, 1, 4) -> Output: INVALID - Invalid Triangle: (4, 1, 2) -> Output: INVALID

P5. Prefix:

1. Equivalence Partitioning:

- Prefix match: ("abc", "abcdef") -> Output: true - No Prefix match: ("abc", "def") -> Output: false

2. Boundary Value Analysis:

- Empty string prefix: ("", "abc") -> Output: true

- Prefix longer than string: ("abcdef", "abc") -> Output: false

P6. Triangle Classification (Floating Point):

a) Equivalence Classes: - Equilateral Triangle

- Isosceles Triangle

- Scalene Triangle

-

- Right-Angled Triangle
- Non-Triangle (Invalid)
- Non-Positive Input

b) Test Cases:

-
Equilateral Triangle: (5.0, 5.0, 5.0) -> Output: Equilateral

- Isosceles Triangle: (5.0, 5.0, 7.0) -> Output: Isosceles

- Scalene Triangle: (3.0, 4.0, 5.0) -> Output: Scalene

- Right-Angled Triangle: (3.0, 4.0, 5.0) -> Output: Right-Angled

- Non-Triangle: (1.0, 2.0, 3.0) -> Output: Invalid

- Non-Positive Input: (-1.0, 2.0, 3.0) -> Output: Invalid

c) Boundary Test Cases for Scalene:

- (3.0, 4.0, 7.0) -> Output: Scalene

- (3.0, 7.0, 4.0) -> Output: Scalene -

(7.0, 3.0, 4.0) -> Output: Scalene

d) Boundary Test Cases for

Isosceles: - (5.0, 5.0, 7.0) ->

Output: Isosceles

- (5.0, 7.0, 5.0) -> Output:

Isosceles - (7.0, 5.0, 5.0) ->

Output: Isosceles

e) Boundary Test Cases for Equilateral:

- (5.0, 5.0, 5.0) -> Output: Equilateral

-

f) Boundary Test Cases for Right-Angled:

- (3.0, 4.0, 5.0) -> Output: Right-Angled
- (5.0, 12.0, 13.0) -> Output: Right-Angled
- (6.0, 8.0, 10.0) -> Output: Right-Angled

g) Boundary Test Cases for Non-

Triangle: - (1.0, 2.0, 3.0) ->

Output: Invalid

- (2.0, 1.0, 3.0) -> Output:

Invalid - (3.0, 1.0, 2.0) ->

Output: Invalid

h) Test Cases for Non-Positive Input:

(-1.0, 2.0, 3.0) -> Output: Invalid

- (1.0, -2.0, 3.0) -> Output: Invalid
- (1.0, 2.0, -3.0) -> Output: Invalid
- (0.0, 2.0, 3.0) -> Output: Invalid