Lab 8: Software Engineering (IT314)
Vivek Chaudhari
202201294

Q1. Consider a program for determining the previous date. Its input is triple of day, month and year with the following ranges 1 <= month <= 12, 1 <= day <= 31, 1900 <= year <= 2015.The possible output dates would be previous date or invalid date. Design the equivalence class test cases?

Write a set of test cases (i.e., test suite) – specific set of data – to properly test the programs. Your test suite should include both correct and incorrect inputs.

1. Enlist which set of test cases have been identified using Equivalence Partitioning and

Boundary Value Analysis separately.

2. Modify your programs such that it runs, and then execute your test suites on the program.

While executing your input data in a program, check whether the identified expected outcome

(mentioned by you) is correct or not.

The solution of each problem must be given in the format as follows:

Equivalence Partitioning:

Tester Action and Input Data (Day, Month, Year)	**Expected Outcome**
Input: 15, 6, 2010 (Valid) Previous date: 14/6/2010	
Input: 1, 3, 2011 (Valid) Previous date: 28/2/2011	
Input: 32, 5, 2012 (Invalid day) Invalid date	
Input: 31, 13, 2014 (Invalid month) Invalid date	

Boundary Value Analysis:

Tester Action and Input Data (Day, Month, Year)	Expected Outcome
Input: 1, 1, 2010 (Boundary case) Previous date: 31/12/2009	
Input: 29, 2, 2012 (Leap year boundary case) Previous date: 28/2/2012	
Input: 31, 12, 2015 (Boundary case) Previous date: 30/12/2015	
Input: 1, 3, 2016 (Leap year boundary case) Previous date: 29/2/2016	

Q2:

P1: Linear Search Program

Equivalence Partitioning:

Tester Action and Input Data (Array, Value)	Expected Outcome
Input: [5, 8, 12, 3], 8 Index 1 Input: [7, 10, 15, 19], 15 Index 2	

Tester Action and Input Data (Array, Value)	Expected Outcome
Input: [1, 2, 3], 4 -1 (not found)	

Boundary Value Analysis:

Tester Action and Input Data (Array, Value)	Expected Outcome
Input: [], 5 (Empty array) -1 (not found)	
Input: [1, 2, 3], 1 (First element) Index 0	
Input: [1, 2, 3], 3 (Last element) Index 2	

P2: Count items program
Equivalence Partitioning:

Tester Action and Input Data (Array, Value) Expected Outcome
Input: [5, 8, 12, 3], 8 1
Input: [7, 10, 15, 7, 7], 7 3
Input: [1, 2, 3], 4 0

Boundary Value Analysis:

Tester Action and Input Data (Array, Value) Expected Outcome
Input: [], 5 (Empty array) 0
Input: [5, 5, 5], 5 3

P3: Binary Search Program
Equivalence Partitioning:

Tester Action and Input Data (Sorted Array, Value) **Expected Outcome**
Input: [3, 6, 8, 12, 15], 8 Index 2
Input: [1, 2, 3, 4], 3 Index 2
Input: [5, 7, 9, 11], 10 -1 (not found)

Boundary Value Analysis:

Tester Action and Input Data (Sorted Array, Value) **Expected Outcome**
Input: [1, 3, 5, 7], 1 (First element) Index 0
Input: [1, 3, 5, 7], 7 (Last element) Index 3
Input: [1, 3, 5, 7], 2 (Not present) -1 (not found)

P4: Triangle Program(Integer Inputs)
Equivalence Partitioning:

Tester Action and Input Data (a, b, c) Expected Outcome
Input: 3, 3, 3 Equilateral triangle
Input: 5, 5, 3 Isosceles triangle
Input: 6, 7, 8 Scalene triangle
Input: 2, 5, 10 Invalid triangle

Boundary Value Analysis:

Tester Action and Input Data (a, b, c) Expected Outcome
Input: 1, 1, 2 Invalid triangle
Input: 3, 4, 5 (Right-angled triangle) Scalene triangle
Input: 6, 6, 10 Isosceles triangle
Input: 2, 2, 4 Invalid triangle

P5: Prefix Program
Equivalence Partitioning:

Tester Action and Input Data (String 1, String 2) **Expected Outcome**
Input: "pre", "prefix" true
Input: "sub", "substring" true
Input: "app", "application" true
Input: "miss", "mismatch" false

Boundary Value Analysis:

Tester Action and Input Data (String 1, String 2)	**Expected Outcome**
Input: "sub", "" (Empty string)	false
Input: "", "substring" (Empty prefix)	true
Input: "longstring", "short" (Longer prefix)	false

P6: Triangle Program (Floating-Point Inputs)

a) Equivalence Classes:

(a) Equilateral Triangle: All sides are equal ($A = B = C$).

(b) Isosceles Triangle: Two sides are equal, and the third is different ($A = B \neq C$, $A \neq B = C$, $A = C \neq B$).

(c Scalene Triangle: All sides are different ($A \neq B \neq C$).

(d Right-Angled Triangle: Satisfies the Pythagorean theorem ($A^2 + B^2 = C^2$). (e Non-Triangle: It cannot form a triangle ($A + B \leq C$, $B + C \leq A$, $C + A \leq B$).

b) Extensive Test Cases:

(a) Equivalence Class: Equilateral Triangle

- Test Case 1: $A = 1$, $B = 1$, $C = 1$ (Minimum positive values)
- Test Case 2: $A = 10$, $B = 10$, $C = 10$ (Larger positive values)

(b) Equivalence Class: Isosceles Triangle

- Test Case 3: $A = 3$, $B = 3$, $C = 4$ ($A = B \neq C$)
- Test Case 4: $A = 4$, $B = 3$, $C = 3$ ($A \neq B = C$)
- Test Case 5: $A = 3$, $B = 4$, $C = 3$ ($A = C \neq B$)

(c) Equivalence Class: Scalene Triangle

- Test Case 6: $A = 3$, $B = 4$, $C = 5$ (Regular scalene triangle)
- Test Case 7: $A = 1$, $B = 2$, $C = 3$ (Smallest positive values)

(d) Equivalence Class: Right-Angled Triangle

- Test Case 8: $A = 3$, $B = 4$, $C = 5$ ($A^2 + B^2 = 9 + 16 = 25 = C^2$)
- Test Case 9: $A = 5$, $B = 12$, $C = 13$ (Another right-angled triangle) (e)

Equivalence Class: Non-Triangle

- Test Case 10: $A = 1$, $B = 2$, $C = 6$ ($A + B = 3 < C$)
- Test Case 11: $A = 0$, $B = 0$, $C = 0$ (All sides are zero)
- Test Case 12: $A = 1$, $B = 1$, $C = 2$ ($A + B = 2 = C$)

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

(a) Test Case 13: $A = 3$, $B = 4$, $C = 6$ ($A + B = 7 > C$)

(b) Test Case 14: $A = 1$, $B = 1$, $C = 2$ ($A + B = 2 < C$)

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

(a) Test Case 15: $A = 5, B = 4, C = 5$ ($A = C$)
(b) Test Case 16: $A = 1, B = 1, C = 2$ ($A \neq C$)

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

(a) Test Case 17: $A = 4, B = 4, C = 4$ ($A = B = C$)

(b) Test Case 18: $A = 1, B = 2, C = 3$ ($A \neq B \neq C$)

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

(a) Test Case 19: $A = 3, B = 4, C = 5$ ($A^2 + B^2 = 9 + 16 = 25 = C^2$)

(b) Test Case 20: $A = 7, B = 24, C = 25$ (Another right-angled triangle)or Non-Triangle Case (Boundary Exploration):

(c) Test Case 21: $A = 1, B = 2, C = 3$ ($A + B = 3 < C$)

(d) Test Case 22: $A = 0, B = 0, C = 1$ (A and B are zero, $A + B = 0 < C$) (e)

Test Case 23: $A = 1, B = 1, C = 3$ ($A + B = 2 < C$)

g) For Non-Positive Input (Boundary Exploration):

(a) Test Case 24: $A = -1, B = 2, C = 3$ (A is non-positive)

(b) Test Case 25: $A = 1, B = -2, C = 3$ (B is non-positive)

(c) Test Case 26: $A = 1, B = 2, C = -3$ (C is non-positive)

(d) Test Case 27: $A = 0, B = 2, C = 3$ (A is zero)

(e) Test Case 28: $A = 1, B = 0, C = 3$ (B is zero)

(f) Test Case 29: $A = 1, B = 2, C = 0$ (C is zero)