Name- Raj kumar
Roll No- 224CS3022
Subject- software Testing
Assignment-1

Q1: Ans: Equivalance classes:

case1: 0 to 120 calls: The bill should be Rs. 300.
example: 105 calls
→ Expected Bill Rs 300

case2: 121 to 190 calls: The bill is Rs 300 + Rs 1 for each call beyond 120
example: 150 calls:
→ Expected Bill ⇒ 300 + (150 − 120) * 1 = 300 + 30 = 330 Ans

case3: 191 to 240 calls: The bill is Rs 300 + Rs 70 + Rs 0.80 for each call beyond 240
examples: 200 calls.
→ Expected Bill ⇒ ∴ 300 + 70 + (200 − 190) * 0.8 = 370 + 8 = ₹ 378 Ans

case4: More than 240 calls: The bill is Rs 300 + 70 + 40 + Rs 0.40 for each call beyond 240
examples ⇒ : 260 calls
→ Expected Bill ⇒ ∴ 300 + 70 + 40 + (260 − 240) * 0.4
= 410 + 20 x 0.4
= 410 + 8 ⇒ RS 418 Ans

cases ⇒ : Invalid Equivalance classes:

- Negetive calls: calls < 0
- Non Integer calls: calls = 120.5

Q2 Ans:

Key considerations:

- Inputs: Three agles of a tringle
- Outputs: Type of tringle (acute, Obtuse angle, right)

Case1: Acute angled tringle

Inputs: 60, 60, 60 → Angles:

output: Acute angled tringle

Reason: All angles are less then 90 degree and their sum is 180 degr

case2: Obtuse Angled tringle

Inputs: Angle → 120, 30, 30

Expected output: Obtuse Angled tringle

Reason: One angle is greater then 90° and their sum is 180°

case3: Right Angled tringle:

Input: Angle 90, 45, 45

Expected ouptut: Right Angle tringle

Reason: One angle is 90° and their sum is 180°

case4: Invalid Tringle (sum Not 180)

→ Input: 60, 60, 70

→ Expected output: Error or invalid tringle

– Reason: The sum of the angles is not 180°

case5: Invalid Tringle (Negative Angle)

→ Input: -60, 120, 120

→ Expected output: Error or Invalid tringle

→ Reason: Angles cannot be negative

case6: Invalid Tringle (zero Angle or two Right Angle)

→ Input: 0, 90, 90

→ Expected output → Error or Invalid tringle

→ Reason → Angles cannot be zero.

Q3: Ans: Equivalen classes:

$$ax^2+bx+c$$

$$x = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

case 1: $b^2 = 4ac$, inputs are $a=3, b=6, c=3$

case 2: $b^2 > 4ac$, input are $b=7, a=2, c=4$

case 3: $b^2 < 4ac$, input are $b=3, a=5, c=6$

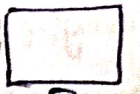
case 4: Invalid Equation inputs are $a=0, b=0$, and $c=10.0$

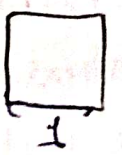
So Test suite = {(3,6,3), (2,7,4), (5,3,6), (0,0,10)}

Q4 Answer: Key consideration:

Inputs: Two pairs of coordinates representing the lower left and upper right corners of two rectangles.

output: Points of intersection of the two rectangles:

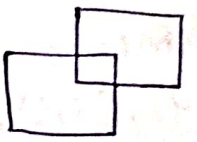




Case 1: No Intersection

- Rectangle 1: (0,0), (2,2)
- Rectangle 2: (3,3), (5,5)
- Expected output: No Intersection points
- Reason: The rectangles are completely separate:

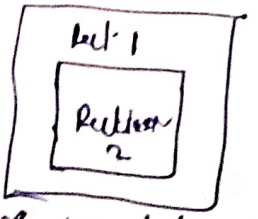
Case 2: Partial Overlap



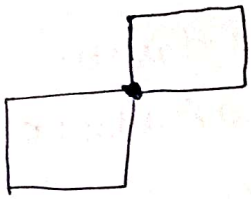
- Rectangle 1: (0,0), (3,3)
- Rectangle 2: (2,2), (4,4)
- expected output: Partial Overlap → Intersection points (2,2), (3,3)
- Reason: The rectangles are partial Overlap

Case 3: One Rectangle Inside Another

- Rectangle 1: (0,0), (5,5)
- Rectangle 2: (1,1) (4,4)
- Expected output: (1,1) (4,4)
- Reason: The second rectangle completly inside the first



Case 4: Touch at a corner



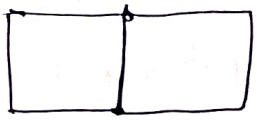
Rectangle 1: (0,0), (2,2)

Rectangle 2: (2,2), (4,4)

Expected output: No Intersection point ~~(2,2)~~

Reason: ~~Both~~ Rectangle ~~Intersect~~ touch at (2,2) Point at single corner

Case 5: Touch at an Edge



Rectangle 1: (0,0), (3,3)

Rectangle 2: (3,0) (5,3)

expected output: No Intersection points ~~(3,0) (3,3)~~

Reason: The rectangle touch along an edge but do not overlap

Case 6: Identical Rectangles

Rectangle 1: (0,0), (2,2)

Rectangle 2: (0,0), (2,2)

- ~~Output~~: Intersecton points (1,1), (3,3)

Reason: The rectangles are identical

Case 7: Overlapping along One Axis:

Rectangle 1: (0,0), (4,2)

Rectangle 2: (2,1) (6,3)

Exp. output: Intersection points (2,1), (4,2)

Reason: The rectangles overlap along the X-axis

Case 8: Invalid rectangles:

① Non-numeric values for coordinates:

- (0,0), (4,4), (5,5), ('x', '4') → Non numeric coordinates are provide

② Improperly defined rectangles

~~Examples~~: (2,2), (0,0), (~~3,3~~), (3,3) Lower-left and upper-right Points are swapped for the first rectangle.

Os: Key considerations:
- Input: Both strings have a size of less than 20 characters.
- Functionality: Check if the first string is substring of the second and count occurrences.

Case 1: Basic substring
String 1: "abc"
String 2: "abcabcabc"
Expected output "abc" occurs 3 times in "abcabcabc"

case 2: No occurrences:
string 1: "xyz"
String 2: "abcdef"
Expected output: "xyz" does not occurs in "abcdef"

case 3: Single character substring
- String 1: "a"
- String 2: "banana" } "a" occurs 3 times in "banana"

case 4: Entire String match
String 1: "hello" } "hello" is exactly the same as the second string.
String 2: "hello"

case 5: overlapping substrings:
string 1: "aa"
string 2: "aaaa" } "aa" occurs 3 times in "aaaa" (overlapping)

case 6: Empty first string:
string 1: "" }
String 2: "abcdef" } An empty string is not considered a valid substring

case 7: Empty second string:
string 1: "abc" } "abc" cannot occurs in an empty string.
string 2: ""

case 8: substring is case sensitive
string 1: "abc"
String 2: "ABCabc" } "abc" occurse 1 time in "ABCabc"

case 9: Invalid case:
① string 1 = NULL
String 2 = "abcdef" } first string is null
② string 1 = "abcdef"
String 2 = "NULL" } second string is null

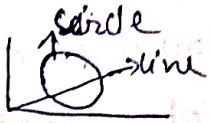
Q6:

Inputs: ① Two points defining a straight line
② A point and a float number defining the center and radius of a circle.

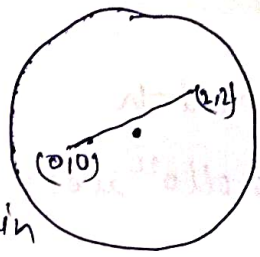
case1: No Intersection
Line: (0,0) and (1,1)
circle: center (4,4) Radius 1 } The line is far from the circle and does not intersect

case2: Tangent to circle:
Line: Point (0,1) and (2,1)
circle: center (1,0), Radius 1
The line is tangent to the circle at one point.

case3: Two Intersection point:
Line: Points (0,0) and (2,2)
Circle: center (1,1) Radius $\sqrt{2}$
The line passes through the circle, intersecting at two points.

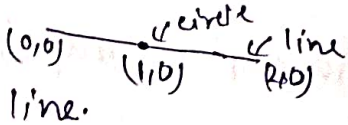


case4: Line inside circle
Line: (0,0) and (2,2)
circle: center (1,1) Radius 5
The entire line segment is within the circle.



case5: line as Diameter
Line: (-1,0) and (1,0)
Circle: center (0,0) Radius 1

case6: Coincident line and circle center
Line: Point (0,0) and (2,0)
circle: center (1,0), Radius 0
The line is a point on the line.



case7: One Intersection point between line and circle
line: Point (3,3) and 5,5)
circle: center (2,2) radius $2\sqrt{2}$
The line passes through the circle and intersects the circle.

case8: Invalid case
→ line definition is Incomplete
line: Point (0,0)
circle: center (1,1) Radius 1 } Does not given the second point of line.

Age: An integer between 30 and 80.
Degree: An integer (0, 1, or 2) representing education level
Wealth: A number between 1 and 10 indicating wealthiness.
output: An integer indicating predicted years to live.

1. Age:
Valid: 30 to 80
Invalid: Below 30, Above 80
2. Degree:
- Valid: 0 (school), 1 (undergraduate), 2 (post graduate)
- Invalid: Any number not in {0, 1, 2},
3. Wealth:
- Valid: 1 to 10
- Invalid: Wealth < 0, Wealth > 10

Case 1: Valid inputs (middle range)
Age: 50
Degree: 1
Wealth: 5
} All inputs are within valid ranges so a valid integer indicating predicted years to live.

Case 2: minimum Age
Age: 30
Degree: 0
Wealth: 3
} Testing boundary condition for age
→ A valid integer indicating predicted years to live.

Case 3: maximum age:
Age: 80
Degree: 2
Wealth: 8
expected output: A valid integer indicating predicted years to live.

Case 4: Invalid Age (Below Range
Age: 25
Degree: 1
Wealth: 5
Expected output: Error or Exception

cases: Invalid Degree:
-Age: 40
Degree: 3
wealth: 6
Expected output: Error or Exception

6. Invalid wealth:
- Age: 60
  Degree: 2
  wealth: 11
  Expected output: Error or exception

Case 7: Valid Input (Edge case:
- Age: 30
  Degree: 2
  wealth: 10
  Expected output: A valid integer indicating predicted years to live
~~Cases~~: