ST mini Project On
Pythagoras Theorem for Right Angle Triangle
2021-22

From:- Uday Kumar J B 1SJ19IS118
Tharun P C 1SJ19IS117
S N Rakesh 1SJ19IS093
VI B ISE

To:-Prof. YOGARAJA G S R
Assistant Professor
Dept. of ISE, SJCIT

Contents:-

1.INTRODUCTION

1.Introduction:-

The problem of Pythagoras theorem for right angled triangle has three inputs and can be tested by writing different test cases in a particular Range (1<inputs<50). Here we going to give the value of AB , BC and CA and we check that given triplet is a right angle or not. AB**2=BC**2+CA**2 In this problem, we are taking only right angle triangle because Pythagoras theorem is only applicable for right angled triangle . For example AB, BC and CA Of a triangle are 5,4 and 3 the given is a right angled triangle else the test cases fail and finally conclude the program has some error . By using the input and range of a input we check the end result vs the expected output by preparing test cases for the given program. Last but not List, In Software Testing we are doing testing on functional testing type this testing is done to increase the performance and to reduce the exception or errors, bugs, etc. We have a formula to find the number of test cases present in given range and the formula is 4b+1, where b be input and conditions . This is based on 3 types of analysis like Boundary value analysis, Decision table of the given Right angle triangle. 1.1Conditions:- Taking three inputs i.e AB,BC,CA from user . Based on the below mentioned condition the Pythagoras theorem helps in helps in forming a right angle triangle or not a Right angle triangle is given as follows:- C1: AB!=BC C2:BC!=CA C3:CA!=AB C4:$AB^2=BC^2+CA^2$ C5:AB>BC AB>C

2.SOFTWARE SPECIFICATIONS

2.Software specifications:-

- OS : Windows 7,8,9,10,11.
- RAM:mini 5GB.
- IDE : Eclipse, any Python IDE.
- ROM: mini 20GB.
- System type: Laptops, System etc.

3.REQUIRMENTS

3.Requirements:

R1. The system should accept 3 positive integer numbers (AB, BC, AC) which represents 3 sides of the Right angle triangle.
R2. Based on the input it should determine what is possibility or output of Right angle triangle.
R3. If the requirement R1 satisfied then the system should determine the type of output in Right angle triangle problem, can be
(I) Right angle triangle (AB**2=BC**2+AC**2)
(II) Not a Right angle triangle (AB**2 !=BC**2+AC**2)
(III)Invalid input (AB=BC or BC=CA or CA=AB)
(IIII)Out of range (1<=AB=BC=CA<=50)
R4. Upper Limit for the size of any sides is 50
. R5.Lower limit for the size of any side is 1.

4.ALGORITHM

4.Algorithm:

- Step1: Input AB , BC and CA three integer values which represents three sides of the Right angle triangle.
- Step2: If (1<=AB<=50 and 1<=BC<=50 and 1<=CA<=50);
- Step3: If (AB**2=BC**2+AC**2): Then Print Right angle triangle. Do step 6.
- Step4: (AB=BC or BC=CA or CA=AB): Then Print invalid input. Do step 6.
- Step 5: else: print not a right angle triangle.
- Step6: Stop. Print out of range.

5.PROGRAM CODE

5.Program Code:

```
AB=int(input("Enter the value of side AB:"))
BC=int(input("Enter the value of side BC: "))
CA=int(input("Enter the value of side CA: "))
if 1<=AB<=50 and 1<=BC<=50 and 1<=CA<=50 :
   if (AB*2==BC2+CA*2):
         print ("Right angle triangle")
    elif (AB==BC or BC==CA or CA==AB):
          print ("Invalid input")
   else:
         print ("Not a right angle triangle")
else:
          print ("Out of range”)
```

6.TESTING INFORMATION

6.1BOUNDARY VALUE ANALYSIS

6.Testing Information:

6.1 Boundary value analysis:-

For BVA problem the test cases can be generation depends on the output. Here we least worried on the constraints on input domain. The right angle triangle problem or Pythagoras theorem for triangle problem takes 3sides as input and check if for validity, hence n=3. Since BVA yields (4n+1) test cases according to single fault assumption theory , hence we say that total number of test cases will be (4*3+1)=13. The maximum limit of each sides AB, BC and CA of the right angle triangle is 50 units according to requirements R4 and lower limit of 1unit.So AB , BC and CA lies between 1<=AB<=50 1<=BC<=50 1<=CA<=50 .

A function F of 2 variables X1 and X2 .When implemented as a program will have same boundaries. A<=X1<=B,C<=X2<=D strongly typed languages permit explicit definition of such variables range. The boundary value analysis test cases are obtained by holding the values of all but are variable at their normal values.

6.2 EQUIVALENCE CLASS

6.2 Equivalence Class:-

They form a partition of a set where partition refers to a collection of mutually disjoint subset union of which is the entire set. This as two implication entire set. This as two implication of testing:

->Entire set represented provides a form completeness.
->Disjointedness ensures a form of non redundancy.

Equivalence class partitioning technique focus on the input domain, we can obtain a richer set of test cases. What are some possibilities for the three integers AB, BC and CA ? they canal be Right angle triangle, not Right angle triangle , Invalid input, out of range. The maximum limit of each sides AB, BC and CA of the right angle triangle is 50 units according to requirement R4 and 1 units according to requirement R4 and 1 units to the requirement R5. So AB, BC and lies between 1<=AB<=50 1<=BC<=50 1<=CA<=50

6.3DECISION TABLE

6.3 Decision Table:-

Decision table have been used to represent and analyse complex logical relationship.

4 Portions of decision table:

(I)The part to the left of the bold vertical line is the stub portion.

(II)To the right is entry portion.

(III)The part above the bold horizontal line is the condition.
(Iv) Portion and below is the action portion.

Decision table based testing has been around since he early 1960's. It is used to depict complex logical relationships between input data. A decision table is method used to build a complete set of test cases without using the internal structure of the program in question. In order to create test cases we use a table to contain the input and output values of program. The '_ _' symbol in the table indicates don't care values. The table shows the 5 conditions and 4 actions. All the conditions in the decision table are binary value. Hence it is called as binary value . Hence it is called as Limited entry decision table.

7.Test cases Tables

7.1 BOUNDARY VALUE TEST CASES

7.1 Boundary value test cases

7.1BOUNDARY VALUE TEST CASES:

TC	INPUTS AB	BC	CA	EXPECTED OUTPUT	ACTUAL OUTPUT	STATUS
1	5	4	3	Right angle triangle	Right angle triangle	Pass
2	13	12	5	Right angle triangle	Right angle triangle	Pass
3	8	6	5	Not a Right angle triangle	Not a Right angle triangle	Pass
4	4	3	2	Right angle triangle	Not a Right angle triangle	Fail
5	25	24	7	Right angle triangle	Right angle triangle	Pass
6	17	15	8	Right angle triangle	Right angle triangle	Pass
7	9	8	7	Right angle triangle	Not a Right angle triangle	Fail
8	41	40	9	Right angle triangle	Right angle triangle	Pass
9	15	14	12	Right angle triangle	Not a Right angle triangle	Fail
10	30	20	10	Not a Right angle triangle	Not a Right angle triangle	Pass
11	12	10	10	Invalid input	Invalid input	Pass
12	8	8	8	Invalid input	Invalid input	Pass
13	10	10	12	Not a Right angle triangle	Invalid input	Fail

7.2 Equivalence Class testing

7.2.1weak Normal Form

7.2.1WEAK NORMAL FORM:

TC	INPUTS AB	BC	CA	EXPECTED OUTPUT	ACTUAL OUTPUT	STATUS
WN1	5	4	3	Right angle triangle	Right angle triangle	Pass
WN2	8	6	5	Not a Right Angle triangle	Not a Right angle triangle	Pass
WN3	12	10	10	Invalid input	Invalid input	Pass
WN4	10	10	12	Invalid input	Invalid input	Pass
WN5	8	8	8	Invalid input	Invalid input	Pass

7.2.2 weak Robust equivalence class testing

7.2.2WEAK ROBUST EQUIVALENCE CLASS TESTING:

TC	INPUTS AB	BC	CA	EXPECTED OUTPUT	STATUS
WR1	-5	4	3	Value of AB not in the range	Pass
WR2	5	-4	3	Value of BC not in the range	Pass
WR3	5	4	-3	Value of CA not in the range	Pass
WR4	53	45	28	Value of AB not in the range	Pass
WR5	45	53	28	Value of BC not in the range	Pass
WR6	45	28	53	Value of AB not in the range	Pass

7.2.3 Strong Robust equivalence class testing

7.2.3STRONG ROBUST EQUIVALENCE CLASS TEST CASES:

TC	INPUTS			EXPECTED OUTPUT	STATUS
	AB	BC	CA		
SR1	-5	4	3	Value of AB not in the range	Pass
SR2	5	-4	3	Value of BC not in the range	Pass
SR3	5	4	-3	Value of CA not in the range	Pass
SR4	-5	-4	3	Value of AB and BC not in the range	Pass
SR5	5	-4	-3	Value of BC and CA not in the range	Pass
SR6	-5	4	-3	Value of AB and CA not in the range	Pass
SR7	-5	-4	-3	Value of AB,BC and CA not in the range	Pass

7.3 DECISION TABLE

7.3 Decision Table

7.3Decision table:-

TC ID	INPUTS AB	BC	CA	EXPECTED OUTPUT	STATUS
DT 1	5	4	3	Right angle triangle	Pass
DT 2	6	5	4	Not a Right angle triangle	Pass
DT 3	9	8	7	Not a Right angle triangle	Pass
DT 4	8	8	8	Invalid input	Pass
DT 5	?	?	?	Impossible	Pass
DT 6	?	?	?	Impossible	Pass
DT 7	13	12	5	Right angle triangle	Pass
DT 8	?	?	?	Impossible	Pass
DT 9	12	10	10	Invalid input	Pass
DT 10	17	8	15	Right angle triangle	Pass
DT 11	10	10	12	Invalid input	Pass

8.Execution and Result Discussion:

Execute the program against the designed actual output columns and status column.

9.Test Report:-

1.Number of TC Executed:42

2.Number of Defects raised:4

3.Number of TC’S passed: 38

4.Number of TC’S failed:4

10.conclusion:

- Our project provide an easy way to analysis the Right angle triangle or not.
- Our project helps in programmatically determine the Right angle triangle or not.
- This system is user friendly and accurate.
- It is more efficient than mannual.

11.references:

- W3schools. com
- Javatpoint. com
- Greeks fot Greeks. com

Thanks you