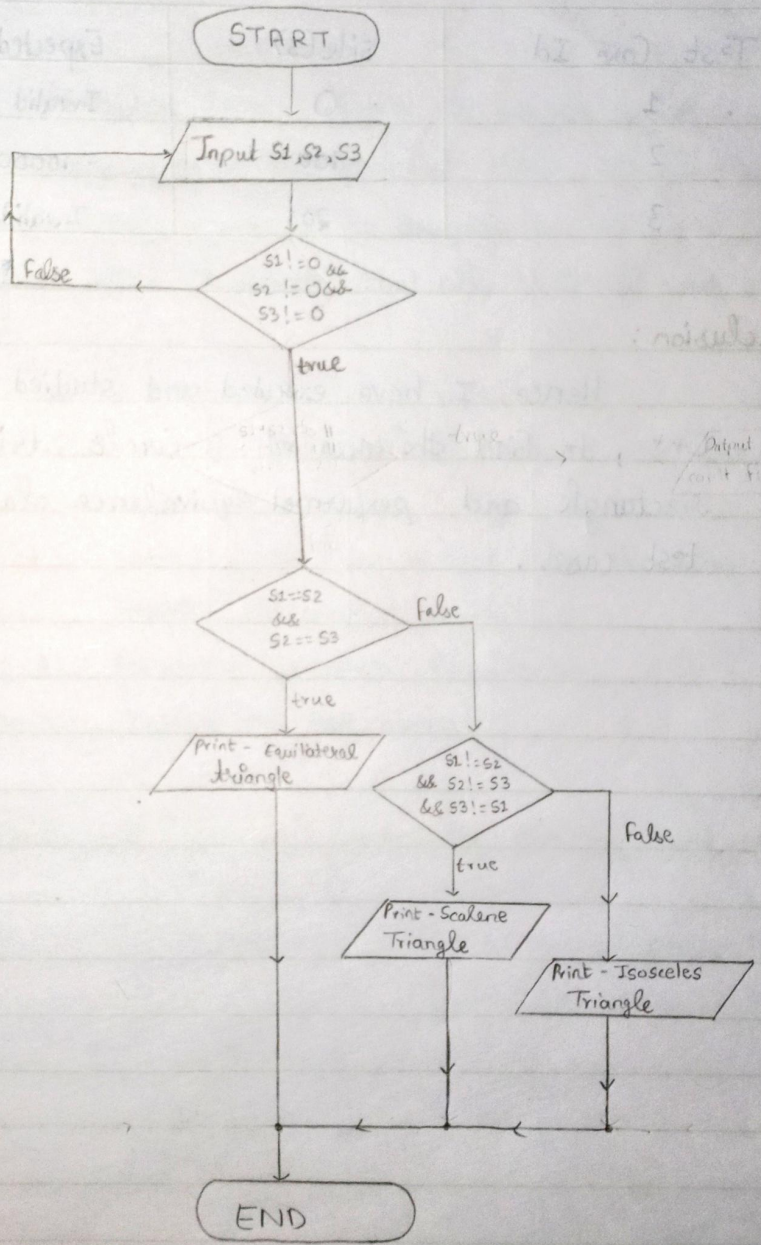
Practical No. 04

Aim: Write a program in C/C++ to read 3 sides of a triangle & to determine whether they form scalene, isosceles or equilateral triangle and test the same thing using basic path testing and find its V(G) by all the three methods

Flowchart:



Practical No. 04

PAGE NO.
DATE

Aim: Write a program in C/C++ to read 3 sides of a triangle and to determine whether they form scalene, isosceles or equilateral triangle and test the same using basis path testing and find its is V(G) by all the three methods.

Theory:

Basis path testing helps a tester to compute logical complexity measures, V(G), of the code. This value of V(G), defines the maximum number of test cases to be designed by identifying basis set of execution paths to ensure that all statements are executed at least once.

Steps to compute the complexity measure, V(G) are as under

Step-1: Construct the flow graph from the source code or flow charts.
Step-2: Identify independent graph.
Step-3: Calculate Cyclomatic Complexity.
Step-4: Design the test cases.

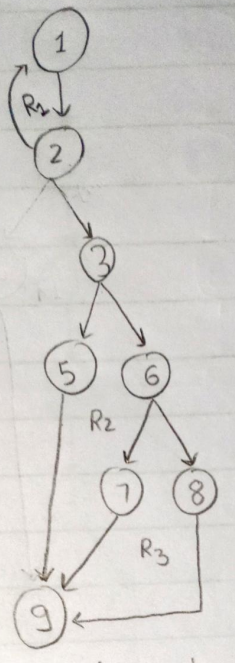
Calculation of cyclomatic complexity V(G) by three methods:

Method-1: $V(G) = E - N + 2$ (E = No. of EDGES) (N = NO. of Nodes)

$V(G) = 11 - 9 + 2$

$V(G) = 4$

Graph:



Directed Graph

1 (Outer Region)

N: Number of Nodes = 9

E: Number of Edges = 11

D: Decision point = 3

	Index
R_1	Region 1
R_2	Region 2
R_3	Region 3

PAGE NO.
DATE

Method - 2 :

$V(G) = P + 1$ (P - No. of predicate nodes with out degree = 2)

$V(G) = 3 + 1$ (Nodes 2, 3, 6 are predecate nodes with two outgoing edges.)

$\boxed{V(G) = 4}$

Method-3:

$V(G)$ = Number of enclosed regions + 1

$= 3 + 1$

$\boxed{V(G) = 4}$ (Here, R_1, R_2, R_3 are the enclosed regions and 1 corresponds to the outer region.)

$\therefore$ $V(G) = 4$ is same by all the three methods.

The test cases for each path are:

	Test case	Valid Input	Expected Results
1	Enlist 1st path	a,b,c : valid input	if a=b or b=c or a=c, then message "isosceles triangle" is displayed.
2	Enlist 2nd path	a,b,c : valid input	if $a \neq b \neq c$ then message 'Scalene triangle" is displayed.
3	Enlist 3rd path	a,b,c : valid input	if a=b=c, message, 'Equilateral triangle" is is displayed.
4	Enlist 4th path	a,b,c : invalid path	Go to, enter values of a, b, c.

Conclusion:

Thus we have studied and executed program to read 3 sides of a triangle and to determine whether they form scalene, isosceles or equilateral triangle and tested the same using basis path testing calculated V(G) by all the three methods.

PAGE NO.
DATE

Code:

```
#include <iostream>
using namespace std;

int main(){
    top:
    Cout << " Input Sides of triangle : \n";
    int s1,s2,s3;
    cout << " Input side 1 : ";
    cin >> s1;
    Cout << " Input side 2 : ";
    cin >> s2;
    Cout << " Input side 3 : ";
    cin >> s3;

    if( s1 != 0 && s2 != 0 && s3 != 0){
        if(s1 == s2 && s2 == s3){
            cout << " The given sides are of an equilateral triangle ";
        } else if( s1 != s2 && s2 != s3 && s1 != s3) {
            cout << " The given sides are of an scalene triangle ";
        }else {
            cout << " The given sides are of an isosceles triangle ";
        }
    }else {
        goto top;
    }
}
```

PAGE NO.
DATE

Conclusion :

Thus we have studied and executed program to read 3 sides of a triangle & to determine whether they form scalene, isosceles, or equilateral triangle and tested the same using basis path testing calculated V(G) by all the three methods.