# SOFTWARE TESTING LAB

# ASSIGNMENT 3

Submitted by:
Shiv Kumar
2016UIT2563

Practical : Write a program to generate test cases for the triangle problem using strong robust equivalence class testing

```
#include<bits/stdc++.h>

using namespace std;

string output(vector<float> &input,float lw[],float up[]){
	string s = "ABC",ans = "";
	for(int i = 0;i<3;i++){
		if(lw[i]>input[i]||up[i]<input[i]){
		ans+=s[i];
		ans+=" ";
		}
	}

	if(ans!=""){
	return "INVALID INPUT AT "+ans;
	}

	if(input[0]>input[1]+input[2] || input[2]>input[1]+input[0] || input[1]>input[0]+input[2])
		return "NOT A TRIANGLE";
	if(input[0]==input[1]&&input[1]==input[2])
		return "EQUILATERAL TRIANGLE";
	if(input[0]==input[1] || input[1]==input[2] || input[2]==input[0])
		return "ISOSCELES TRIANGLE";

	return "SCALENE TRIANGLE";
}


int main(){
	string s = "ABC";
	float lw[3],up[3];
	for(int i = 0;i<3;i++){
		while(1){
			cout<<"Enter valid range of "<<s[i]<<" (LOWER UPPER) : ";
			cin>>lw[i]>>up[i];
			if(lw[i]<=up[i]) break;
			}
	}

	cout<<"2 NONVALID and 1 VALID FOR EACH VARIABLE"<<endl;

	vector<float> in[3];

	for(int i = 0;i<3;i++){
		in[i].push_back(lw[i]-1);
		in[i].push_back((lw[i]+up[i])/2);
		in[i].push_back(up[i]+1);
	}
```

```
        string sp5 = " ",sp10 = sp5+sp5;
        string sp4 = " ",sp8 = sp4+sp4;
        cout<<sp5<<"A"<<sp10<<"B"<<sp10<<"C"<<sp10<<sp10<<"EXPECTED
OUTPUT"<<endl;
        for(int i = 0;i<3;i++){
                for(int j = 0;j<3;j++){
                        for(int k = 0;k<3;k++){
                                cout<<setw(6)<<in[0][i];
                                cout<<setw(11)<<in[1][j];
                                cout<<setw(11)<<in[2][k];
                                cout<<sp10<<sp5;
                                vector<float> input;
                                input.push_back(in[0][i]);
                                input.push_back(in[1][j]);
                                input.push_back(in[2][k]);
                                cout<<output(input,lw,up)<<endl;
                        }
                }
        }

        return 0;
}
```

**output:**

```
(base) Shivs-Air:Software Testing championballer$ subl prac9.cpp
(base) Shivs-Air:Software Testing championballer$ g++ prac9.cpp -o run
(base) Shivs-Air:Software Testing championballer$ ./run
Enter valid range of A (LOWER UPPER) : 1 100
Enter valid range of B (LOWER UPPER) : 1 100
Enter valid range of C (LOWER UPPER) : 1 100
2 NONVALID and 1 VALID FOR EACH VARIABLE
 A  B  C    EXPECTED OUTPUT
     0          0          0   INVALID INPUT AT A B C
     0          0       50.5   INVALID INPUT AT A B
     0          0        101   INVALID INPUT AT A B C
     0       50.5          0   INVALID INPUT AT A C
     0       50.5       50.5   INVALID INPUT AT A
     0       50.5        101   INVALID INPUT AT A C
     0        101          0   INVALID INPUT AT A B C
     0        101       50.5   INVALID INPUT AT A B
     0        101        101   INVALID INPUT AT A B C
  50.5          0          0   INVALID INPUT AT B C
  50.5          0       50.5   INVALID INPUT AT B
  50.5          0        101   INVALID INPUT AT B C
  50.5       50.5          0   INVALID INPUT AT C
  50.5       50.5       50.5   EQUILATERAL TRIANGLE
  50.5       50.5        101   INVALID INPUT AT C
  50.5        101          0   INVALID INPUT AT B C
  50.5        101       50.5   INVALID INPUT AT B
  50.5        101        101   INVALID INPUT AT B C
   101          0          0   INVALID INPUT AT A B C
   101          0       50.5   INVALID INPUT AT A B
   101          0        101   INVALID INPUT AT A B C
   101       50.5          0   INVALID INPUT AT A C
   101       50.5       50.5   INVALID INPUT AT A
   101       50.5        101   INVALID INPUT AT A C
   101        101          0   INVALID INPUT AT A B C
   101        101       50.5   INVALID INPUT AT A B
   101        101        101   INVALID INPUT AT A B C
```