```
//============================================================================
// Name        : Assignment6.cpp
// Author      : Chatura Ahangama
// Version     :
// Copyright   : This is private property
// Description : Hello World in C++, Ansi-style
//============================================================================

#include <iostream>
#include <cmath>

using namespace std;

class shape{
        protected:
                double area;   //Class Protected Data member to store Area
                double perimeter; //Class Protected Data member to store perimeter

        public:
                shape(){};//Default Constructor is declared for style
                ~shape(){};
                virtual  void calcPerimeter()=0;   //Pure Virtual Base
                virtual  void calcArea()=0;             //Pure Virtual Base

};

class rectangle:public shape{    //Class instance to create rectangle
        public:
            rectangle(double leng,double wid)//Default Constructor takes in the length and width
                {
                length=leng; //Initializes to the internal length and width
                width=wid;
                calcPerimeter();    //Calls the member function to evaluate the Perimeter
                calcArea();  //Calls the member function local to rectangle
                };


                void calcPerimeter(){   //Calculates the Perimeter
                        perimeter=(length*2)+(width*2);  //Adds the sides together
                        Lperi=perimeter; //Declares this private member that only my friend can access but client can't
access
                };


                void calcArea(){  //Calculates the Area
                        area=length*width;  //Multiplies together to get the area
                        Larea=area; //Declares this private member that only my friend can acess but client cant acess
                };

            private:

                double length; //Length is relevant type for the rectangle
                double width;  //Width is relevant for the rectangle
                double Lperi;  //Local that the friend can acess
                double Larea; //friend cant acess any private friends that you inherit

                friend void printPerimeter(rectangle);
                friend void printArea(rectangle);

};



void printPerimeter(rectangle IN)  //Takes in as a friend
{

        cout<<"\nPerimeter is :"<<IN.Lperi<<endl; //Access the necessary Lpreri from rectangle

}

void printArea(rectangle IN)
{

        cout<<"\nArea is :" <<IN.Larea<<endl; //Access the necessary area

}

class triangle:public shape{ //Declaration for the triangle class
        public:
                triangle(double s1,double s2,double s3){  //Deafult constructor takes the three sides
                        side1=s1;
                        side2=s2;
                        side3=s3;
                        calcPerimeter();  //Calculates the Perimeter
                        calcArea(); //Calculates the Parameter
                };

                void calcPerimeter(){
                        perimeter=side1+side2+side3;  //Add all the sides
                        Lperi=perimeter;  //set the permeter equal the local Lperi mainly for friend to come print
                };

                void calcArea(){ //calculates the Area
            double PP=perimeter/2;  //Alorythm states P=perimeter/2;
                        double t1=(PP)*(PP-side1)*(PP-side2)*(PP-side3);   //P(P-s1)(P-s2)(P-s3) = T1
                        area=sqrt(t1); //Square root of t1 gives the area
                        Larea=area;   //Assigns its place holder till it gets called by the friend function
                };


        private:
                double side1; //Private members only member functions and friends can access
                double side2;
```

```
                double side3;
                double Lperi; //Specifically for the friends to come grab
                double Larea; //Specifically for the friends to come grab

                friend void printPerimeter(triangle); //friendly to any calls for a print function
                friend void printArea(triangle); //friendly to any calls for a print function

};


void printPerimeter(triangle IN)
{
        cout<<"\nPerimeter is :"<<IN.Lperi<<endl; //Acess the class private member Lperi

}


void printArea(triangle IN)
{

        cout<<"\nArea is :" <<IN.Larea<<endl;  //Acess the class private member Larea

}



int main()
{

        rectangle test(30,20);  // Instance of a Rectangle
        triangle  test1(20,29,40); //Instance of a triangle

        cout << "\nRunning Tringle Test Cases :: " << endl; // prints Test Case
        printArea(test1);
    printPerimeter(test1);

    cout << "\nEnds Tringle Test Cases :: " << endl; // prints Test Case


    cout << "\nRunning Rectangle Test Cases :: " << endl; // prints Test Case
    printArea(test);
    printPerimeter(test);

    cout << "\nEnds Tringle Test Cases :: " << endl; // prints Test Case


        return 0;
}
```