```
/*
  File: triangles.cpp
  Created by: Tan Qi Hao
  Creation Date: 4/11/2019
  Synopsis: This program represent an axis-aligned right triangle in the
xy
  plane and output the coordinates of each vertexes, dimensions,
hypotenuse and
  perimeter. Then, it will output each of the aspect again under scale
factor in
  x and y direction.
*/

#include <iostream>
#include <cmath>

using namespace std;

class Point
{
private:
      double px;
      double py;

public:
      void setX(const double x);
      void setY(const double y);
      double getX() const;
      double getY() const;
};

class Triangle
{
private:
      Point blPoint;
      double length, height;

public:
      // member functions
      void setBottomLeftX(const double x);
      void setBottomLeftY(const double y);
      void setLength(const double inLength);
      void setHeight(const double inHeight);

      Point getBottomLeft() const;
      Point getBottomRight() const;
      Point getTopLeft() const;
      double getLength() const;
      double getHeight() const;

      double perimeter() const;
      double hypotenuse() const;
      void scaleLength(const double sx);
      void scaleHeight(const double sy);
```

```
      void display() const;
};

// FUNCTION PROTOTYPES GO HERE:
void read_triangle(Triangle & tri);

int main()
{
      // Define local variables
      Triangle tri;
      double sx, sy;

      //Prompt the user for triangle information and fill Class Triangle
object, tri,
      //with this information
      read_triangle(tri);

      // Display triangle information
      tri.display();

      // Prompt and read scale factors to change length and height
      cout << "Enter scale factor in x direction: ";
      cin >> sx;

      cout << "Enter scale factor in y direction: ";
      cin >> sy;

      // Apply scale factors
      tri.scaleLength(sx);
      tri.scaleHeight(sy);

      // Display triangle information
      tri.display();

      return 0;
}

// FUNCTION DEFINITIONS GO HERE:

// CLASS MEMBER FUNCTION DEFINITINOS GO HERE:

void Point::setX(const double x)
{
      px = x;
}

void Point::setY(const double y)
{
      py = y;
}

double Point::getX() const
{
      return (px);
```

```
}

double Point::getY() const
{
      return (py);
}

void Triangle::setBottomLeftX(const double x)
{
  blPoint.setX(x);
}

void Triangle::setBottomLeftY(const double y)
{
  blPoint.setY(y);
}

void Triangle::setLength(const double inLength)
{
  length = inLength;
}

void Triangle::setHeight(const double inHeight)
{
  height = inHeight;
}

Point Triangle::getBottomLeft() const
{
  return blPoint;
}

Point Triangle::getBottomRight() const
{
  Point Botright; //Define a variable under Point class data type

  double x = blPoint.getX() + length;
  Botright.setX(x);
  return Botright;

}

Point Triangle::getTopLeft() const
{
  Point Topleft; //Variable Topleft under point data type.

  double y = blPoint.getY() + height;
  Topleft.setY(y);
  return Topleft;

}

double Triangle::getLength() const
{
```

```
  return length;
}

double Triangle::getHeight() const
{
  return height;
}

double Triangle::hypotenuse() const
{
  return sqrt(height * height + length * length);
}

double Triangle::perimeter() const
{
  return sqrt(height * height + length * length) + height + length;
}

void Triangle::scaleLength(const double scalefact)
{
  length = scalefact * length;
}

void Triangle::scaleHeight(const double scalefact)
{
  height = scalefact * height;
}

void Triangle::display() const
{
  cout << "-----------------------------------------------" << endl;

  cout << "Lower Left Vertex (" << blPoint.getX() << ", "
       << blPoint.getY() << ") "<< endl;

  cout << "Top Left Vertex (" << blPoint.getX() << ", "
       << getTopLeft().getY() << ") "<< endl;

  cout << "Bottom Right Vertex (" << getBottomRight().getX() << ", "
       << blPoint.getY() << ") "<< endl;

  cout << "Dimensions (" << getBottomRight().getX() -  blPoint.getX() <<
", "
       <<  getTopLeft().getY() - blPoint.getY() << ") "<< endl;


 cout << "Hypotenuse = " << hypotenuse() << endl;

 cout << "Perimeter = " << perimeter() << endl;

  cout << "-----------------------------------------------" << endl;
}

void read_triangle(Triangle & tri)
```

```
{

  double x, y; //variable x and y coordinate
  double length, height; //variable length and height dimension


      cout << "Enter bottom left x coordinate: ";
      cin >> x;
      cout << "Enter bottom left y coordinate: ";
      cin >> y;
      cout << "Enter length: ";
      cin >> length;
      cout << "Enter height: ";
      cin >> height;

      cout << endl;

      tri.setBottomLeftX(x);
        tri.setBottomLeftY(y);
      tri.setLength(length);
      tri.setHeight(height);


}
```