```
    1: // Copyright 2022 Anson Cheang
    2: /**
    3: * Triangle.cpp - as an implementation to create a new triangle object to
store every point
    4: and draw out the triangle at a moments notice
    5: *
    6: * Date 2/22/22 - 2/28/22
    7: *
    8: * Created by: Anson Cheang
    9: *
   10: */
   11: #include "Triangle.h"
   12: #include <string>
   13: #include <cstdlib>
   14: #include <iostream>
   15: #include <cmath>
   16: #include <SFML/System.hpp>
   17: #include <SFML/Window.hpp>
   18: #include <SFML/Graphics.hpp>
   19:
   20: // using namespace std;
   21:
   22:
   23: Triangle::Triangle(double val, sf::Vector2f position, char color) {
   24:     size = val;
   25:     sf::Vector2f point1 = position, point2 = position;
   26:     point1.x = point1.x + val;
   27:     point2.x = (position.x + point1.x)/2;
   28:     point2.y = point2.y + sqrt(3)/2 * val;
   29:     P1 = position;
   30:     P2 = point1;
   31:     P3 = point2;
   32:     shape.setPointCount(3);
   33:     shape.setPoint(0, position);
   34:     shape.setPoint(1, point1);
   35:     shape.setPoint(2, point2);
   36:     shape.setOutlineThickness(1);
   37:     if (color == 'g') {
   38:         shape.setOutlineColor(sf::Color::Green);
   39:     } else if (color == 'r') {
   40:         shape.setOutlineColor(sf::Color::Red);
   41:     } else if (color == 'b') {
   42:         shape.setOutlineColor(sf::Color::Blue);
   43:     } else {
   44:         shape.setOutlineColor(sf::Color::Black);
   45:     }
   46: }
   47:
   48: sf::Vector2f Triangle::getP1() {
   49:     return P1;
   50: }
   51:
   52: sf::Vector2f Triangle::getP2() {
   53:     return P2;
   54: }
   55:
   56: sf::Vector2f Triangle::getP3() {
   57:     return P3;
   58: }
   59:
   60: void Triangle::draw(sf::RenderTarget& target, sf::RenderStates states) co
nst {
   61:     target.draw(shape, states);
   62: }
```