```
    1: // Angel Zheondre Calcano
    2: #include <SFML/Graphics.hpp>
    3: #include <SFML/Window.hpp>
    4: #include <cmath>
    5: #include <math.h>
    6: #include <iostream>
    7: #include "sierpinski.hpp"
    8:
    9: using namespace sf ;
   10: using namespace std ;
   11:
   12: int Sierpinski( double side, double depth, double p2x, double p2y, sf::Rende
rWindow& a ) {
   13:
   14:   if(depth == 0) return 0 ;
   15:
   16:   double p0y, p0x, p1y, p1x, mp1x, mp1y, mp2x, mp2y, mp3x, mp3y ;
   17:
   18:   double ylnd = (.866)*side ;
   19:
   20:   //p0's values, left point inner triangle
   21:   p0y = p2y - ylnd ;
   22:   p0x = p2x - side*(.5) ;
   23:
   24:   //p1's values, right point inner triangle
   25:   p1y = p2y - ylnd ;
   26:   p1x = p2x + side*(.5) ;
   27:
   28:   //Draw picture of triangle
   29:   ConvexShape T ;
   30:   T.setPointCount(3) ;
   31:   T.setPoint(0, Vector2f(p2x,p2y));
   32:   T.setPoint(1, Vector2f(p1x,p1y));
   33:   T.setPoint(2, Vector2f(p0x,p0y));
   34:   T.setPosition(0,0);
   35:   a.draw(T) ;
   36:
   37:   // Find the midpoints of the other triangles
   38:   mp1x = p2x + side/2 ;
   39:   mp1y = p2y ;
   40:   mp2x = p2x ;
   41:   mp2y = p0y ;
   42:   mp3x = p2x - side/2 ;
   43:   mp3y = p2y  ;
   44:
   45:   Sierpinski( side/2 , depth - 1, mp1x, mp1y, a ) ;
   46:   Sierpinski( side/2 , depth - 1, mp2x, mp2y, a ) ;
   47:   Sierpinski( side/2 , depth - 1, mp3x, mp3y, a ) ;
   48:
   49:   return 0 ;
   50: }
```