**Practical no:10 B**

**Aim**: Implementation of fractal generation- Sirepenski Triangle

**Theory:**

**The Sierpiński triangle (sometimes spelled Sierpinski), also called the Sierpiński gasket or Sierpiński sieve, is a fractal attractive fixed set with the overall shape of an equilateral triangle, subdivided recursively into smaller equilateral triangles. Originally constructed as a curve, this is one of the basic examples of self-similar sets—that is, it is a mathematically generated pattern that is reproducible at any magnification or reduction. It is named after the Polish mathematician Wacław Sierpiński, but appeared as a decorative pattern many centuries before the work of Sierpiński. The concept behind this is the fact that the filled triangle is filled by an empty equilateral triangle in the center in such a way that this triangular space is congruent to the three triangles being formed around it.**

**Code:**

#include<iostream.h>

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

#include<math.h>

#include<dos.h>

void cons(float x1,float y1,float x2,float y2,float x3,float y3)

{float x4,x5,x6,y4,y5,y6;

x4=(x1+x2)/2; y4=(y2+y1)/2;

x5=(x3+x1)/2; y5=(y3+y1)/2;

x6=(x3+x2)/2; y6=(y3+y2)/2;

line(x4,y4,x5,y5);

line(x5,y5,x6,y6);

line(x6,y6,x4,y4);

if(abs(x4-x5)>0&&abs(x5-x6)>0&&abs(x6-x4)>0)

{delay(2);

cons(x1,y1,x4,y4,x5,y5);

cons(x4,y4,x2,y2,x6,y6);

cons(x5,y5,x6,y6,x3,y3);

}

}

void main() {

clrscr();

int sx,sy;

int gd=DETECT,gm;

initgraph(&gd,&gm,"c:\\tc\\bgi");

float x1,x2,x3,y1,y2,y3;

x1=getmaxx()/2; y1=0;

x2=0; y2=getmaxy();

x3=getmaxx(); y3=getmaxy();

float x4,y4,x5,y5,x6,y6;

x4=(x1+x2)/2; y4=(y2+y1)/2;

x5=(x3+x1)/2; y5=(y3+y1)/2;

x6=(x3+x2)/2; y6=(y3+y2)/2;

line(x1,y1,x2,y2);

line(x1,y1,x3,y3);

line(x3,y3,x2,y2);

line(x4,y4,x5,y5);

line(x5,y5,x6,y6);

line(x6,y6,x4,y4);

cons(x1,y1,x4,y4,x5,y5);

cons(x4,y4,x2,y2,x6,y6);

cons(x5,y5,x6,y6,x3,y3);

getch();

} 