//draw pattern

#include <iostream.h>

#include <stdlib.h>

#include <graphics.h>

#include<math.h>

#include<dos.h>

#include<conio.h>

//using namespace std;

class Line

{

public:

void dda(int x1,int y1,int x2,int y2)

{

   float x,y,dx,dy,length,xincr,yincr;

   int i;

   dx = abs(x2-x1);

   dy = abs(y2-y1);

   if(dx>dy)

      {length = dx;}

   else

      {length = dy;}

   xincr= (x2-x1)/(float)length;

   yincr= (y2-y1)/(float)length;

   x = x1 + 0.5;

   y = y1 + 0.5;

   for(i=1;i<=length;i++)

   {

      putpixel(x,y,12);

      x = x + xincr;

      y = y + yincr;

      }

    putpixel(x,y,12);

 } //end of DDA function

}; //end of clas Line

class Circle

{

public:

void drawcircle(float x1,float y1,int r)

{

    float x,y,d;

    d = 3 - (2 \* r);

    x=0;

    y=r;

    while(x<=y)

    {

       plot(x,y,x1,y1);

       if(d<0)

   {

     x=x+1;

     d = d + (4 \* x) + 6;

    }

       else

   {

     x=x+1;

     y=y-1;

     d = d + (4 \* (x - y)) + 10;

   }

}

}

void plot(int x,int y,int x1,int y1)

{

  putpixel(x+x1,y+y1,12);

  putpixel(y+x1,x+y1,12);

  putpixel(-x+x1,y+y1,12);

  putpixel(y+x1,-x+y1,12);

  putpixel(x+x1,-y+y1,12);

  putpixel(-y+x1,x+y1,12);

  putpixel(-x+x1,-y+y1,12);

  putpixel(-y+x1,-x+y1,12);

}

};

int main()

{

Circle c;

int gd=DETECT,gm;

int x,y,r1,r2,x1,y1,x2,y2,x3,y3;

initgraph(&gd,&gm,"c:\\turboc3\\bgi");

cout<<"Enter centre coordinate (x,y)of circle";

cin>>x>>y;

cout<<"Enter radius ofouter circle ";

cin>>r1;

cout<<"Enter radius of inner circle";

cin>>r2;

c.drawcircle(x,y,r1);

c.drawcircle(x,y,r2);

float length;

length=sqrt((r1\*r1)-(r2\*r2));

x1=x;

y1=y-r1;

x2=x-length;

y2=y+r2;

x3=x+length;

y3=y+r2;

Line obj;

obj.dda(x1,y1,x2,y2);delay(500);

obj.dda(x1,y1,x3,y3);delay(500);

obj.dda(x2,y2,x3,y3);

getch();

closegraph();

system("clear");

return 0;

}