```
#include<bits/stdc++.h>
#include<graphics.h>
using namespace std;

void Bresenhamline(int x1,int y1,int x2,int y2)
{
    int dx,dy,inc1,inc2,inc3,inc4,d;
    float m;

    if(x1>x2)
    {
        int tmp = x1;
        x1 = x2;
        x2 = tmp;
    }
    if(y1>y2)
    {
        int tmp = y1;
        y1 = y2;
        y2 = tmp;
    }

    dx=x2-x1;
    dy= y2-y1;
    m = dy/dx;
    if(m<=1)
    {
        d = 2*dy-dx;
        inc1 = 2*dy;
        inc2 = 2*(dy-dx);
        while(x1<=x2)
        {
            putpixel(x1,y1,15);
            if(d<=0)
            {
                d = d+inc1;
            }
            else{
                d =d+inc2;
                y1++;
            }
            x1++;
        }
    }
    else
    {
        d = 2*dx-dy;
        inc3 = 2*dx;
        inc4 = 2*(dx-dy);
        while(y1<=y2)
        {
            putpixel(x1,y1,15);
            if(d<=0)
            {
                d = d+inc3;
            }
            else{
                d =d+inc4;
                x1++;
            }
            y1++;
        }
    }
}

void BresenhamCircle(int x1,int y1,int r)
{
    int x,y,d;
    x = 0;
    y = r;
    d = 3-(2*r);
    while(x<=y)
    {
        putpixel(x1+x,y1+y,15);
        putpixel(x1-x,y1+y,15);
        putpixel(x1+x,y1-y,15);
        putpixel(x1-x,y1-y,15);
        putpixel(x1+y,y1+x,RED);
        putpixel(x1+y,y1-x,RED);
        putpixel(x1-y,y1+x,RED);
        putpixel(x1-y,y1-x,RED);
        x=x+1;
        if(d<0)
        {
            d = d+4*(x)+6;
        }
        else
        {
            d = d+4*(x-y)+10;
            y=y-1;

        }
```

```c
        delay(20);

    }
}

int main()
{
    int gd=DETECT,gm;
    initgraph(&gd,&gm,"");


    Bresenhamline(10,10,400,11);
    Bresenhamline(10,10,9,250);
    Bresenhamline(9,250,399,249);
    Bresenhamline(399,249,400,11);

    int px1=10,py1=10;
    int px2=9,py2=250;
    int px3=399,py3=249;
    int px4=400,py4=11;
    int cx = round((px1+px4)/2);
    int cy = round((py1+py2)/2);
    BresenhamCircle(cx,cy,60);

    int x1 = (px1+px4)/2;
    int y1 = 10;
    int y2 = cy - 60;
    Bresenhamline(x1,y1,x1+1,y2);

    int ly1 = 250;
    int ly2 = cy + 60;
    Bresenhamline(x1,ly1,x1+1,ly2);


    getch();

//  while(!kbhit())
//  {
//      delay(200);
//  }
    return 0;
}
```