

Assignment No : 01

Name : Gangurde Vaishnavi Bharat

Roll No : 25

Class : SE

Div : A

Problem Statement :

Write a c++ program to draw the following pattern : Use DDA Line & Bresenham's circle algorithms. Apply the concept of encapsulation in it.

Source Code :

```
#include <iostream>
#include <graphics.h>
#include <math.h>
#include <dos.h>

class Algorithm
{
public:
    void DDA_Line(float x1, float y1, float x2, float y2);
    void Bresenham_Circle(int r);
};

// DDA LINE ALGORITHM :
void Algorithm::DDA_Line(float x1, float y1, float x2, float y2)
{
    float x, y, dx, dy, Dx, Dy, length;
    int i, c;
    dx = abs(x2 - x1);
    dy = abs(y2 - y1);
    length = dx >= dy ? dx : dy;
    Dx = (x2 - x1) / length;
    Dy = (y2 - y1) / length;
```

```

x = x1 + 0.5;
y = y1 + 0.5;

i = 1;
while (i <= length)
{
    putpixel(320 + x, 240 - y, 15);
    x = x + Dx;
    y = y + Dy;
    i++;
}
}

// BRESENHAM CIRCLE ALGORITHM:
void Algorithm::Bresenham_Circle(int r)
{
    int x = 0, y = r;
    int d = 3 - (2 * r);
    while (x < y)
    {
        putpixel(320 + x, 240 - y, 15);
        putpixel(320 - x, 240 + y, 15);
        putpixel(320 - x, 240 - y, 15);
        putpixel(320 + y, 240 + x, 15);
        putpixel(320 + y, 240 - x, 15);
        putpixel(320 - y, 240 + x, 15);
        putpixel(320 - y, 240 - x, 15);

        if (d <= 0)
        {
            d = d + (4 * x) + 6;
        }
        else
        {
            d = d + (4 * (x - y)) + 6;
            y = y - 1;
        }

        x = x + 1;
    }
}

```

```
int main()
{
    Algorithm a1;

    int gm, gd = DETECT;
    initgraph(&gd, &gm, "C:\\TC\\BGI");
    int r;

    cout << "Enter the radius" << endl;
    cin >> r;

    // LARGE CIRCLE
    a1.Bresenham_Circle((int)r);

    // TRIANGLE :
    float rad = 3.14 / 180;
    float c = r * cos(30 * rad);
    float s = r * sin(30 * rad);

    a1.DDA_Line(0, r, 0 - c, 0 - s);
    a1.DDA_Line(0, r, 0 + c, 0 - s);
    a1.DDA_Line(0 - c, 0 - s, 0 + c, 0 - s);

    // SMALL CIRCLE:
    float r1 = (sqrt(3) * (r * sqrt(3))) / 6;

    a1.Bresenham_Circle((int)r1);

    delay(3000);
    closegraph();

    return 0;
}
```

OUTPUT :

