Code

```
from OpenGL.GL import *
from OpenGL.GLUT import *
from OpenGL.GLU import *
import math

def draw_points(x, y):
    glPointSize(3) #pixel size. by default 1 thake
    glBegin(GL_POINTS)
    glVertex2f(x, y) #jekhane show korbe pixel
    glEnd()


def circle_points(x, y, X, Y):
    draw_points(X+x, Y+y)
    draw_points(Y+y, X+x)
    draw_points(Y+y, X-x)
    draw_points(X+x, Y-y)
    draw_points(X-x, Y-y)
    draw_points(Y-y, X-x)
    draw_points(Y-y, X+x)
    draw_points(X-x, Y+y)

def midpoint_circle(rad, X, Y):
    d = 1 - rad
    x = 0
    y = rad

    while x <= y:
        circle_points(x, y, X, Y)
        if d < 0:
            d = d + 2 * x + 3
            x += 1
        else:
            d = d + 2 * (x - y) + 5
            x += 1
            y -= 1


def draw_circles(R, X, Y):
```

```
    midpoint_circle(R, X, Y)
    midpoint_circle(R / 2, X + (R/2), Y)
    midpoint_circle(R / 2, X, Y + (R/2))
    midpoint_circle(R / 2, X - (R / 2), Y)
    midpoint_circle(R / 2, X, Y - (R / 2))

    val = math.sin(math.radians(45)) * (R / 2)

    midpoint_circle(R / 2, X + val, Y + val)
    midpoint_circle(R / 2, X + val, Y - val)
    midpoint_circle(R / 2, X - val, Y + val)
    midpoint_circle(R / 2, X - val, Y - val)


def iterate():
    glViewport(0, 0, 500, 500)
    glMatrixMode(GL_PROJECTION)
    glLoadIdentity()
    glOrtho(0.0, 500, 0.0, 500, 0.0, 1.0)
    glMatrixMode(GL_MODELVIEW)
    glLoadIdentity()

def showScreen():
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT)
    glLoadIdentity()
    iterate()
    glColor3f(0.65, 0.25, 0.75) #konokichur color set (RGB)

    # code starts here ================================================
    draw_circles(150, 250, 250)
    glutSwapBuffers()


glutInit()
glutInitDisplayMode(GLUT_RGBA)
glutInitWindowSize(500, 500) #window size
glutInitWindowPosition(0, 0)
wind = glutCreateWindow(b"Lab 3") #window name
glutDisplayFunc(showScreen)
```

glutMainLoop()

Output