**Task3:**

from OpenGL.GL import \*

from OpenGL.GLUT import \*

from OpenGL.GLU import \*

def DDA(x1, y1, x2, y2):

m = (y1-y2)/(x1-x2)

glBegin(GL\_POINTS)

glVertex2f(x1, y1)

while x1<=x2:

x1 = x1+1

y1 = y1+m

glVertex2f(x1, y1)

glEnd()

def DDA\_Sp(x1,y1,x2,y2):

glBegin(GL\_POINTS)

glVertex2f(x1, y1)

while y1 < y2:

y1 = y1 + 2

glVertex2f(x1, y1)

glEnd()

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(1.0, 0.0, 0.0)

#call the draw methods here

DDA(150,400,350,400)

DDA\_Sp(250,100,250,400)

glutSwapBuffers()

glutInit()

glutInitDisplayMode(GLUT\_RGBA)

glutInitWindowSize(500, 500)

glutInitWindowPosition(0, 0)

wind = glutCreateWindow(b"OpenGL Coding Practice")

glutDisplayFunc(showScreen)

glutIdleFunc(showScreen)

glutMainLoop()