K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

Batch: B2 Roll No.: 110

Experiment No. 09

TITLE: WAP to implement Simple Interaction with the mouse and keyboard.

AIM:

Write an OpenGL program to demonstrate use of interaction through mouse and keyboard. (Example: Pressing ‘p’ draws a dot at the mouse position; pressing the left arrow key adds a point to some (global) list, but does no drawing; pressing ‘E’ exits from the program.)

Expected OUTCOME of Experiment:

CO1***Understand the basic concepts of computer graphics and OpenGL***

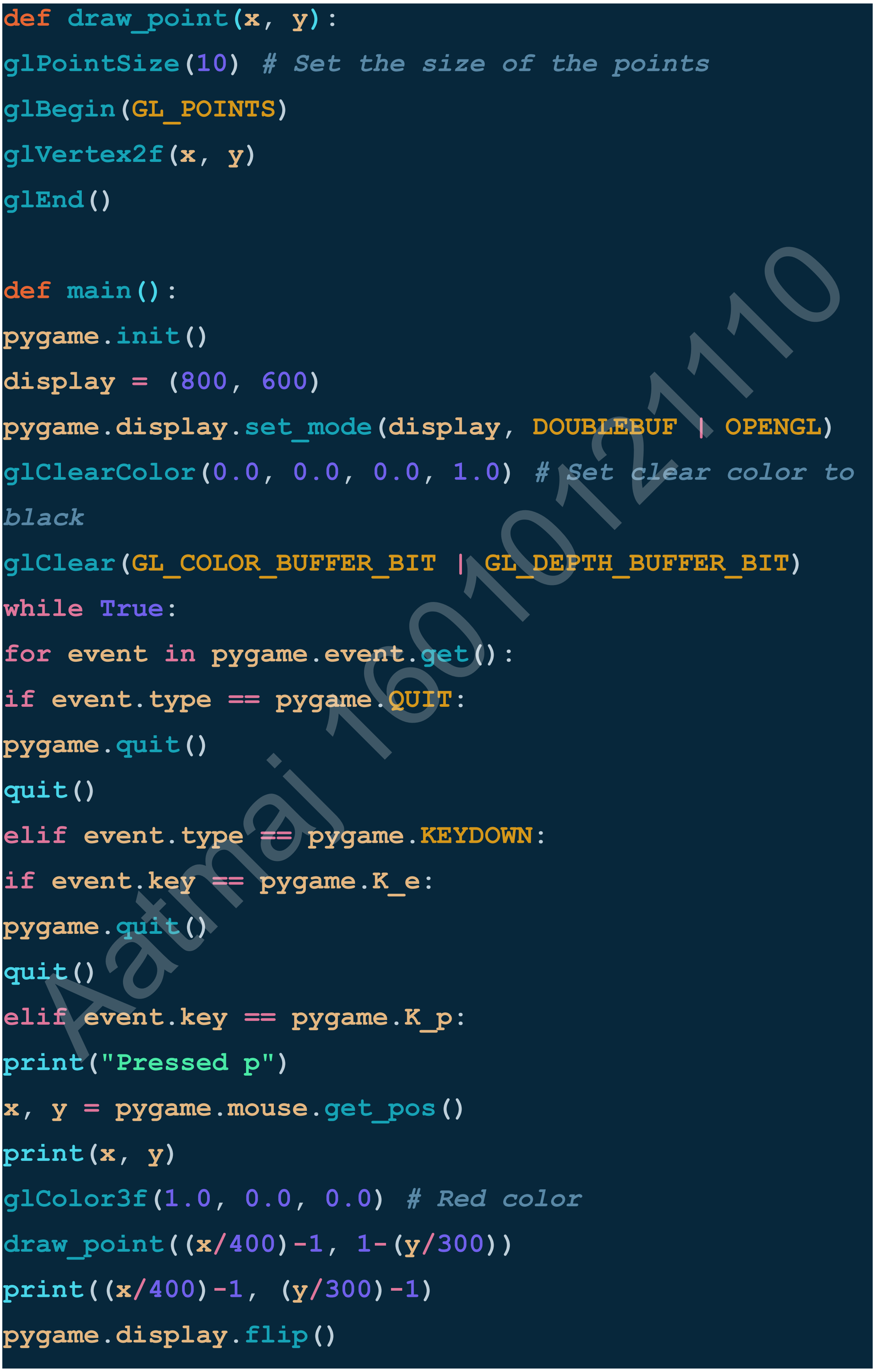
CO4: ***Understand the computer Input& interaction, Curves and Computer Animation***

Books/ Journals/ Websites referred:

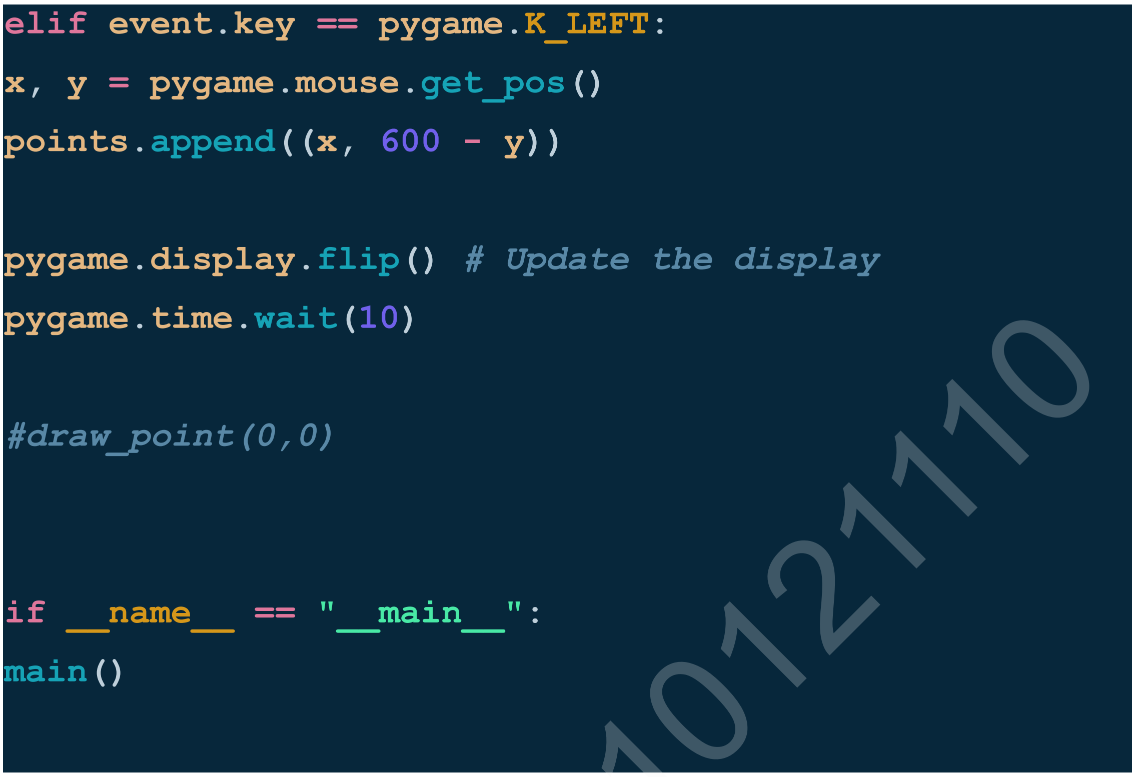
https://chat.openai.com/

Implementation details

```
import pygame
from pygame.locals import *
from OpenGL.GL import *
from OpenGL.GLUT import *

# Global list to store points
points = []

```



```
def draw_point(x, y):
glPointSize(10) # Set the size of the points
glBegin(GL_POINTS)
glVertex2f(x, y)
glEnd()

def main():
pygame.init()
display = (800, 600)
pygame.display.set_mode(display, DOUBLEBUF | OPENGL)
glClearColor(0.0, 0.0, 0.0, 1.0) # Set clear color to
black
glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT)
while True:
for event in pygame.event.get():
if event.type == pygame.QUIT:
pygame.quit()
quit()
elif event.type == pygame.KEYDOWN:
if event.key == pygame.K_e:
pygame.quit()
quit()
elif event.key == pygame.K_p:
print("Pressed p")
x, y = pygame.mouse.get_pos()
print(x, y)
glColor3f(1.0, 0.0, 0.0) # Red color
draw_point((x/400)-1, 1-(y/300))
print((x/400)-1, (y/300)-1)
pygame.display.flip()
```



```
elif event.key == pygame.K_LEFT:
x, y = pygame.mouse.get_pos()
points.append((x, 600 - y))

pygame.display.flip() # Update the display
pygame.time.wait(10)

#draw_point(0,0)


if __name__ == "__main__":
main()
```

Output(s) (Screen Shot):

Conclusion and discussion:

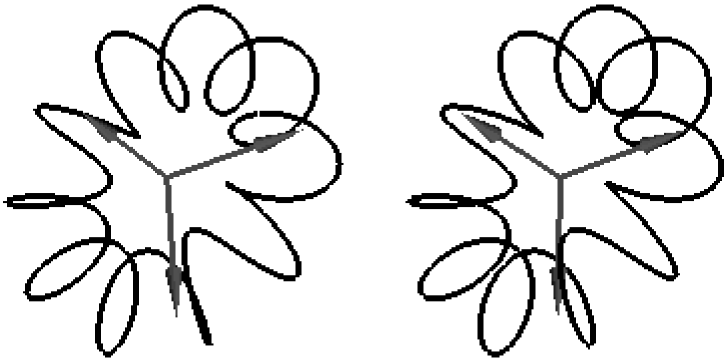
Thus we made a simple interaction based controlling in openGL using pygame. We also made tile design

Date: 4 oct 2023

Signature of faculty in-charge

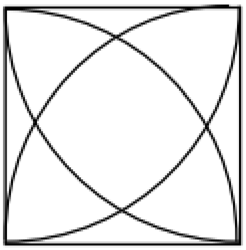
Post lab Question

Write a program to draw the following

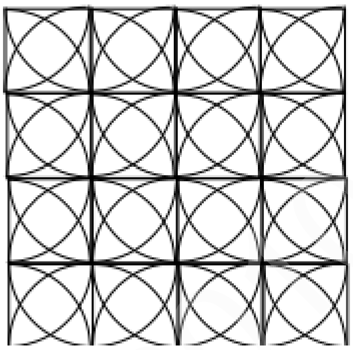


And

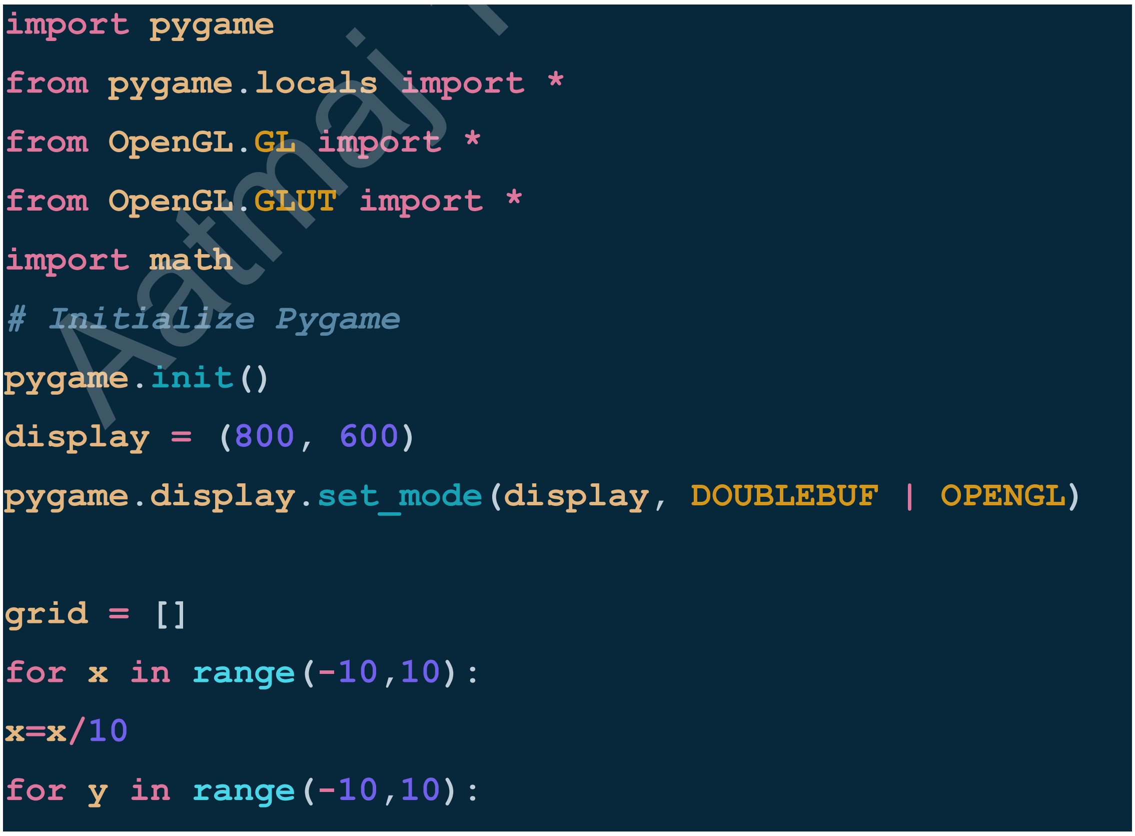
).



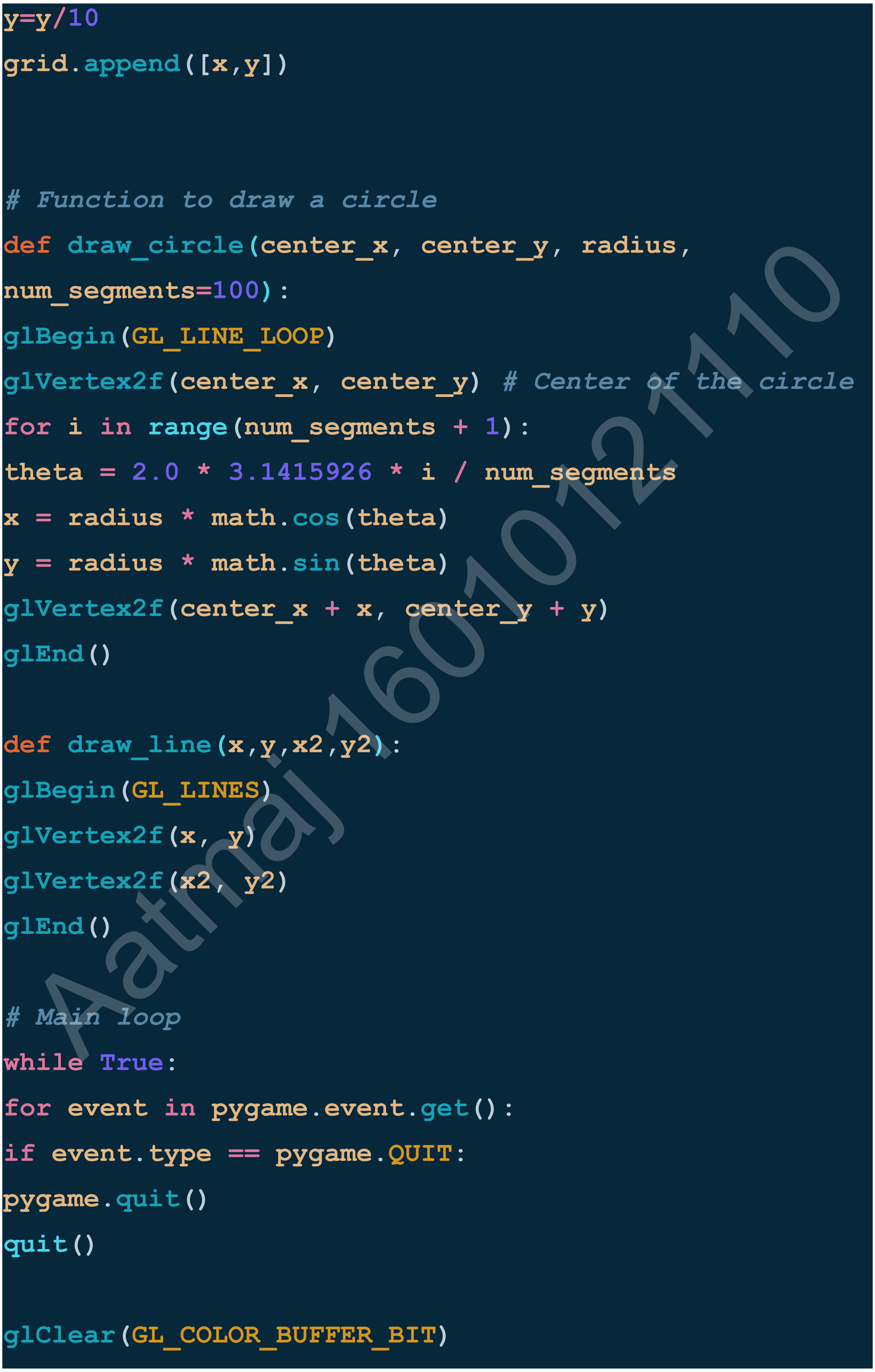
b).



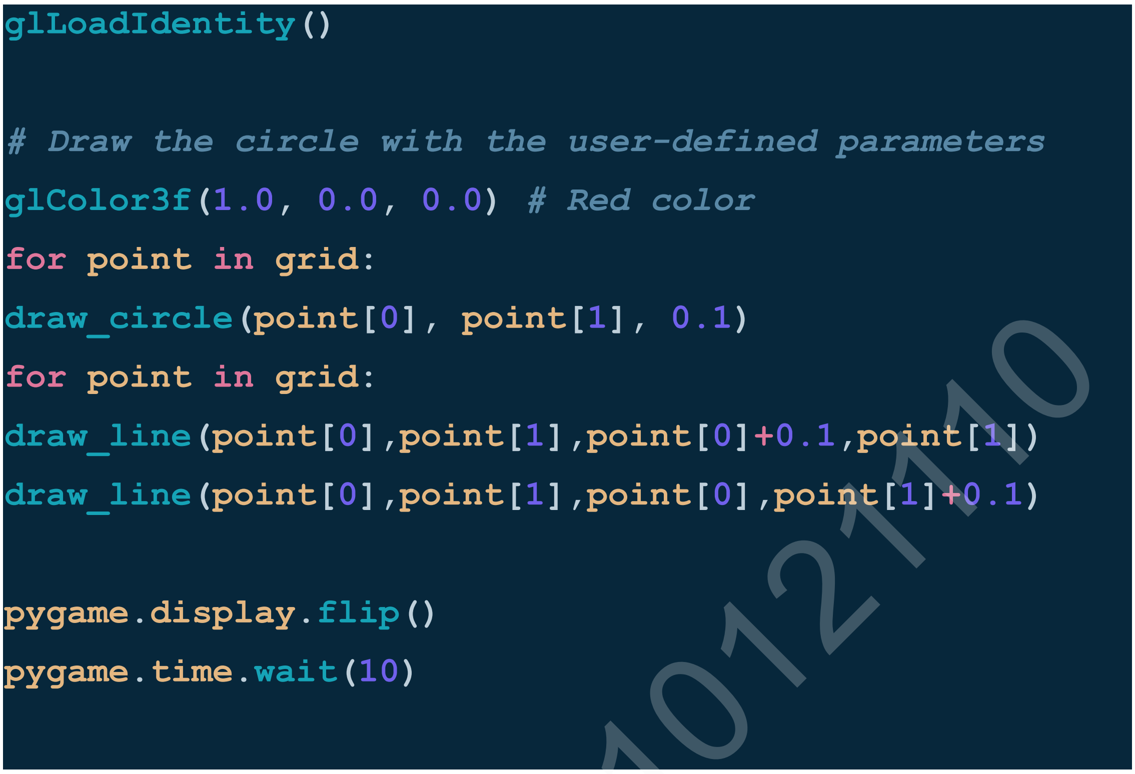
I



```
import pygame
from pygame.locals import *
from OpenGL.GL import *
from OpenGL.GLUT import *
import math
# Initialize Pygame
pygame.init()
display = (800, 600)
pygame.display.set_mode(display, DOUBLEBUF | OPENGL)

grid = []
for x in range(-10,10):
x=x/10
for y in range(-10,10):
```



```
y=y/10
grid.append([x,y])


# Function to draw a circle
def draw_circle(center_x, center_y, radius,
num_segments=100):
glBegin(GL_LINE_LOOP)
glVertex2f(center_x, center_y) # Center of the circle
for i in range(num_segments + 1):
theta = 2.0 * 3.1415926 * i / num_segments
x = radius * math.cos(theta)
y = radius * math.sin(theta)
glVertex2f(center_x + x, center_y + y)
glEnd()

def draw_line(x,y,x2,y2):
glBegin(GL_LINES)
glVertex2f(x, y)
glVertex2f(x2, y2)
glEnd()

# Main loop
while True:
for event in pygame.event.get():
if event.type == pygame.QUIT:
pygame.quit()
quit()

glClear(GL_COLOR_BUFFER_BIT)
```



```
glLoadIdentity()

# Draw the circle with the user-defined parameters
glColor3f(1.0, 0.0, 0.0) # Red color
for point in grid:
draw_circle(point[0], point[1], 0.1)
for point in grid:
draw_line(point[0],point[1],point[0]+0.1,point[1])
draw_line(point[0],point[1],point[0],point[1]+0.1)

pygame.display.flip()
pygame.time.wait(10)
```