**MODUL 4: Ploting Curva**

TUJUAN

1. Membuat kurva 2 dimensi menggunakan openGL

2. Mengarsir area tertentu pada kurva menggunakan openGL

Latihan 4\_1.py

import pygame

from pygame.locals import \*

import numpy as np

import math

from OpenGL.GL import \*

from OpenGL.GLU import \*

from OpenGL.GLUT import \*

**def** init():

    glClearColor(1,0,0, 1);

**def** plotFunc() :

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(3.0)

    glBegin(GL\_LINES)

    glVertex2f(-5.0, 0.0)

    glVertex2f(5.0, 0.0)

    glVertex2f(0.0, 5.0)

    glVertex2f(0.0, -5.0)

    glEnd()

    for x in np.arange(-5.0, 5.0, 0.01):

        y = x\*x

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

*#pygame.time.wait(50)*

        glEnd()

        glFlush()

**def** main():

    pygame.init()

    display = (600,600)

    pygame.display.set\_caption ('Nurul Izza Farhana - 0102523729')

    pygame.display.set\_mode (display, DOUBLEBUF | OPENGL)

    gluPerspective (45, (display [0]/display [1]), 0.1, 50.0)

*# glutInitWindowSize(800, 600);  # Set dimensi window*

    glTranslatef (0.0,0.0, -5)

    glClearColor(1.0, 0.0, 0.0, 1.0)

    init()

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit ()

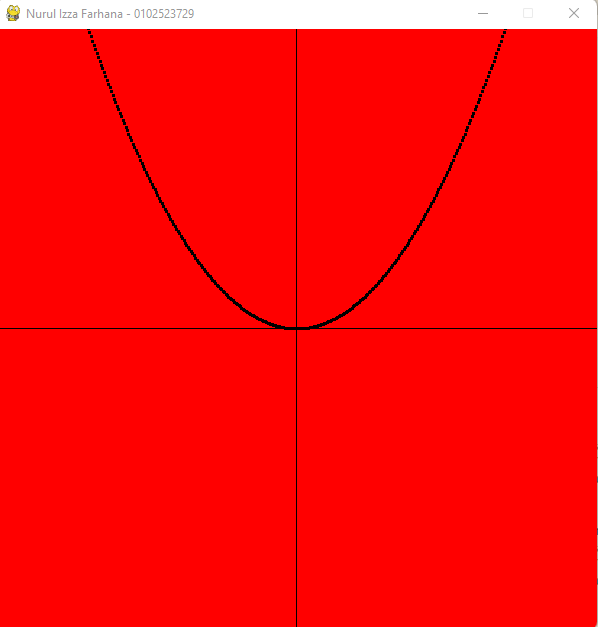
        glClear (GL\_COLOR\_BUFFER\_BIT| GL\_DEPTH\_BUFFER\_BIT)

        plotFunc()

        pygame.display.flip()

        pygame.time.wait (10)

main()



Latihan 4\_2.py

import pygame

from pygame.locals import \*

from OpenGL.GL import \*

from OpenGL.GLU import \*

import numpy as np

import math

**def** plotfunc():

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    for x in np.arange(-5.0, 5.0, 0.01):

        y = x\*x

        glColor3f(0.0, 0.0, 0.0)

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glEnd()

        for a in np.arange(-5.0, 5.0, 0.01):

            if a < x\*x:

                glColor3f(0.50,0.50,0.50)

                glBegin(GL\_POINTS)

                glVertex2f(x,a)

                glEnd()

                glColor3f(0.0, 0.0, 0.0)

    glBegin(GL\_LINES)

    glVertex2f(-5.0, 0.0)

    glVertex2f(5.0, 0.0)

    glVertex2f(0.0, 5.0)

    glVertex2f(0.0, -5.0)

    glEnd()

    glFlush()

**def** plotfunc2():

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(3.0)

    glBegin(GL\_LINES)

    glVertex2f(-5.0, 0.0)

    glVertex2f(5.0, 0.0)

    glVertex2f(0.0, 5.0)

    glVertex2f(0.0, -5.0)

    glEnd()

    for x in np.arange(-5.0, 5.0, 0.01):

        y = x\*x

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

*# pygame.time.wait(50)*

        glEnd()

        glFlush()

**def** init():

    glClearColor (1.0, 0.0, 0.0, 1.0); *## background colour*

**def** main():

    pygame.init()

    display = (400,400)

    pygame.display.set\_caption('Function Plot')

    pygame.display.set\_mode(display, DOUBLEBUF|OPENGL)

    gluPerspective(45, (display[0]/display[1]), 0.1, 50.0)

    glTranslatef(0.0,0.0, -5)

    glClearColor(0, 1.0, 0.0, 1.0)

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit()

        glClear(GL\_COLOR\_BUFFER\_BIT|GL\_DEPTH\_BUFFER\_BIT)

        plotfunc()

        pygame.display.flip()

        pygame.time.wait(10)

main()

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit ()

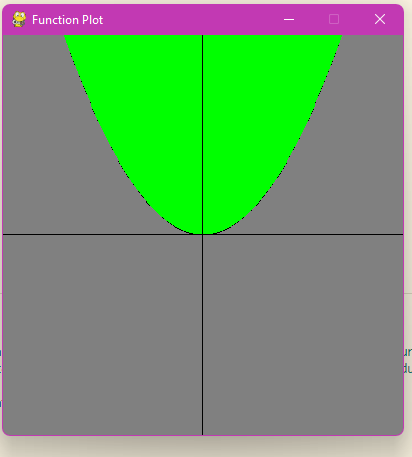
        glClear (GL\_COLOR\_BUFFER\_BIT| GL\_DEPTH\_BUFFER\_BIT)

        plotFunc1()

        pygame.display.flip()

        pygame.time.wait (10)

main()



1. Buatlah program untuk fungsi berikut:  
   a. 𝑦 < 𝑥2 – 2

import pygame

from pygame.locals import \*

import numpy as np

import math

from OpenGL.GL import \*

from OpenGL.GLU import \*

from OpenGL.GLUT import \*

**def** init():

    glClearColor(0,1,0, 1);

**def** plotFunc1() :

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    for x in np.arange(-5.0, 5.0, 0.01):

        y = x\*x - 2

        glColor3f(0.0, 0.0, 0.0)

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glEnd()

        for a in np.arange(-5.0, 5.0, 0.01):

            if a < ((x\*x) - 2):

                glColor3f(0.50,0.50,0.50)

                glBegin(GL\_POINTS)

                glVertex2f(x,a)

                glEnd()

                glColor3f(0.0, 0.0, 0.0)

    glBegin(GL\_LINES)

    glVertex2f(-5.0, 0.0)

    glVertex2f(5.0, 0.0)

    glVertex2f(0.0, 5.0)

    glVertex2f(0.0, -5.0)

    glEnd()

    glFlush()

**def** main():

    pygame.init()

    display = (600,600)

    pygame.display.set\_caption ('Nurul Izza Farhana - 0102523729')

    pygame.display.set\_mode (display, DOUBLEBUF | OPENGL)

    gluPerspective (45, (display [0]/display [1]), 0.1, 50.0)

*# glutInitWindowSize(800, 600);  # Set dimensi window*

    glTranslatef (0.0,0.0, -5)

    init()

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit ()

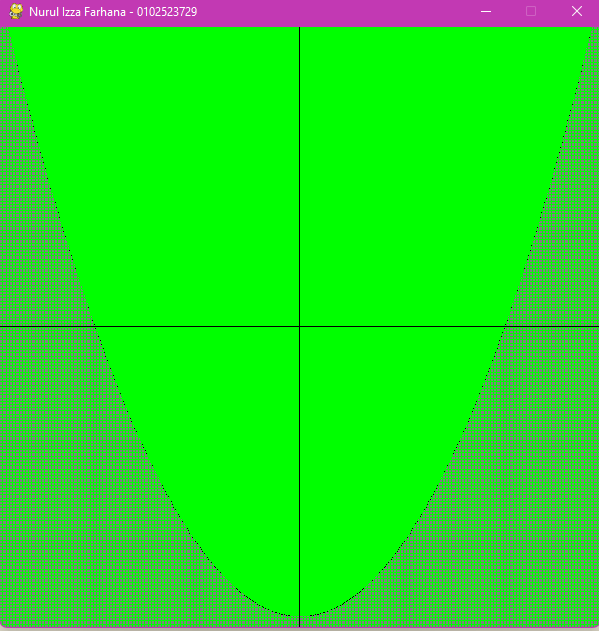
        glClear (GL\_COLOR\_BUFFER\_BIT| GL\_DEPTH\_BUFFER\_BIT)

        plotFunc1()

        pygame.display.flip()

        pygame.time.wait (10)

main()



b. 𝑦 = 𝑥3 - 3𝑥 – 1

import pygame

from pygame.locals import \*

import numpy as np

import math

from OpenGL.GL import \*

from OpenGL.GLU import \*

from OpenGL.GLUT import \*

**def** init():

    glClearColor(0,1,0, 1);

**def** plotFunc2() :

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    for x in np.arange(-5.0, 5.0, 0.01):

        y = x\*x\*x - 3\*x - 1

        glColor3f(0.0, 0.0, 0.0)

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glEnd()

        for a in np.arange(-5.0, 5.0, 0.01):

            if a < (y):

                glColor3f(0.50,0.50,0.50)

                glBegin(GL\_POINTS)

                glVertex2f(x,a)

                glEnd()

                glColor3f(0.0, 0.0, 0.0)

    glBegin(GL\_LINES)

    glVertex2f(-5.0, 0.0)

    glVertex2f(5.0, 0.0)

    glVertex2f(0.0, 5.0)

    glVertex2f(0.0, -5.0)

    glEnd()

    glFlush()

**def** main():

    pygame.init()

    display = (600,600)

    pygame.display.set\_caption ('Nurul Izza Farhana - 0102523729')

    pygame.display.set\_mode (display, DOUBLEBUF | OPENGL)

    gluPerspective (45, (display [0]/display [1]), 0.1, 50.0)

*# glutInitWindowSize(800, 600);  # Set dimensi window*

    glTranslatef (0.0,0.0, -5)

    init()

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit ()

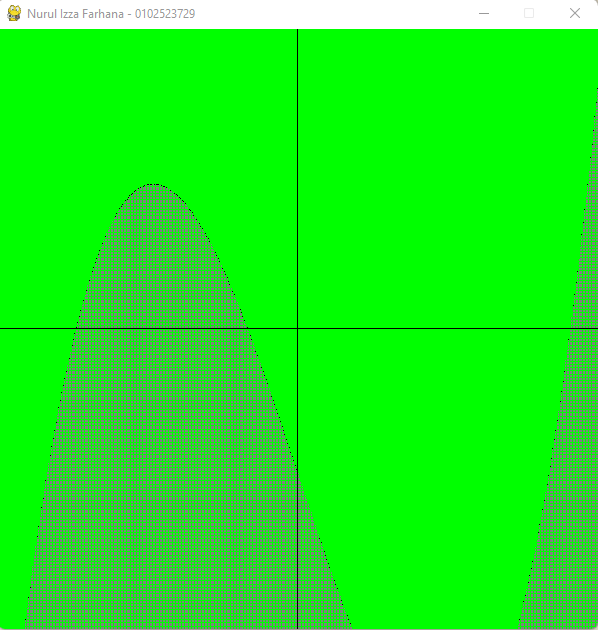
        glClear (GL\_COLOR\_BUFFER\_BIT| GL\_DEPTH\_BUFFER\_BIT)

        plotFunc2()

        pygame.display.flip()

        pygame.time.wait (10)

main()



c. 𝑦 = 𝑠𝑖𝑛(𝑥)

import pygame

from pygame.locals import \*

import numpy as np

import math

from OpenGL.GL import \*

from OpenGL.GLU import \*

from OpenGL.GLUT import \*

**def** init():

    glClearColor(0,1,0, 1);

**def** plotFunc3() :

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    for x in np.arange(-5.0, 5.0, 0.01):

        y = np.sin(x)

        glColor3f(0.0, 0.0, 0.0)

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glEnd()

        for a in np.arange(-5.0, 5.0, 0.01):

            if a < (y):

                glColor3f(0.50,0.50,0.50)

                glBegin(GL\_POINTS)

                glVertex2f(x,a)

                glEnd()

                glColor3f(0.0, 0.0, 0.0)

    glBegin(GL\_LINES)

    glVertex2f(-5.0, 0.0)

    glVertex2f(5.0, 0.0)

    glVertex2f(0.0, 5.0)

    glVertex2f(0.0, -5.0)

    glEnd()

    glFlush()

**def** main():

    pygame.init()

    display = (600,600)

    pygame.display.set\_caption ('Nurul Izza Farhana - 0102523729')

    pygame.display.set\_mode (display, DOUBLEBUF | OPENGL)

    gluPerspective (45, (display [0]/display [1]), 0.1, 50.0)

    glTranslatef (0.0,0.0, -5)

    init()

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit ()

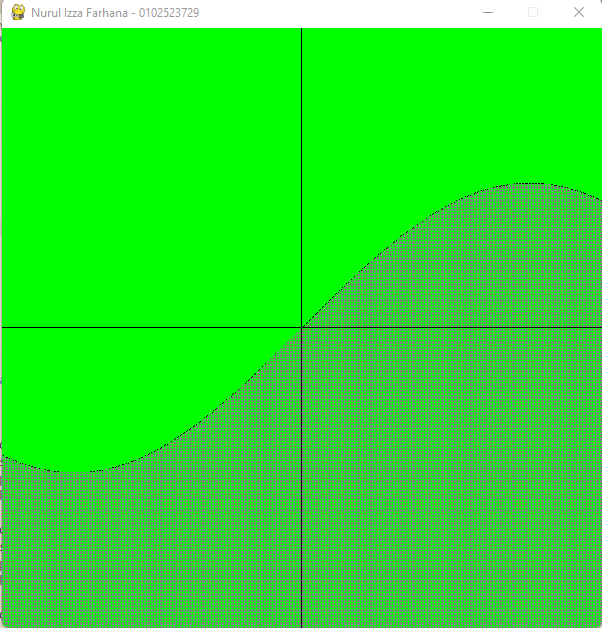
        glClear (GL\_COLOR\_BUFFER\_BIT| GL\_DEPTH\_BUFFER\_BIT)

        plotFunc3()

        pygame.display.flip()

        pygame.time.wait (10)

main()



1. Kesimpulan pada modul 3

Dengan menghubungkan koordinat titik x dan y, kita mampu membentuk kurva di bidang kartesian yang menciptakan garis sebagai representasi visual dari area yang diarsir berdasarkan grafik yang telah dibuat.

**MODUL 5: Plot Fungsi Parametrik**

TUJUAN

1. Pemetaan fungsi parametrik dengan openGL

2. Pembuatan gambar abstrak menggunakan fungsi parametrik dengan openGL

Latihan 5\_1.py Kurva Lingkaran

import pygame

from pygame.locals import \*

from OpenGL.GL import \*

from OpenGL.GLU import \*

import numpy as np

import math

**def** plotfunc():

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    glBegin(GL\_LINES)

    glVertex2f(-2.0, 0.0)

    glVertex2f(2.0, 0.0)

    glVertex2f(0.0, 2.0)

    glVertex2f(0.0, -2.0)

    glEnd()

    for t in np.arange(-5.0,6.28, 0.001):

        x = math.sin(t)

        y = math.cos(t)

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glEnd()

        glFlush()

**def** main():

    pygame.init()

    display = (400,400)

    pygame.display.set\_caption('Function Plot')

    pygame.display.set\_mode(display, DOUBLEBUF|OPENGL)

    gluPerspective(45, (display[0]/display[1]), 0.1, 50.0)

    glTranslatef(0.0,0.0, -5)

    glClearColor(1.0, 1.0, 1.0, 1.0)

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit()

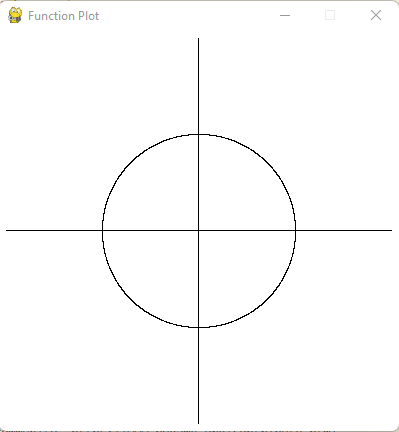
        glClear(GL\_COLOR\_BUFFER\_BIT|GL\_DEPTH\_BUFFER\_BIT)

        plotfunc()

        pygame.display.flip()

        pygame.time.wait(10)

main()



Latihan5\_2.py fungsi garis y=x (memotong pusat lingkaran)

import pygame

from pygame.locals import \*

from OpenGL.GL import \*

from OpenGL.GLU import \*

import numpy as np

import math

**def** plotfunc():

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    glBegin(GL\_LINES)

    glVertex2f(-2.0, 0.0)

    glVertex2f(2.0, 0.0)

    glVertex2f(0.0, 2.0)

    glVertex2f(0.0, -2.0)

    glEnd()

    for t in np.arange(0.0,6.28, 0.001):

        x = math.sin(t)

        y = math.cos(t)

        z = t

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glVertex2f(t,z)

        glEnd()

    glFlush()

**def** main():

    pygame.init()

    display = (400,400)

    pygame.display.set\_caption('Function Plot')

    pygame.display.set\_mode(display, DOUBLEBUF|OPENGL)

    gluPerspective(45, (display[0]/display[1]), 0.1, 50.0)

    glTranslatef(0.0,0.0, -5)

    glClearColor(1.0, 1.0, 1.0, 1.0)

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit()

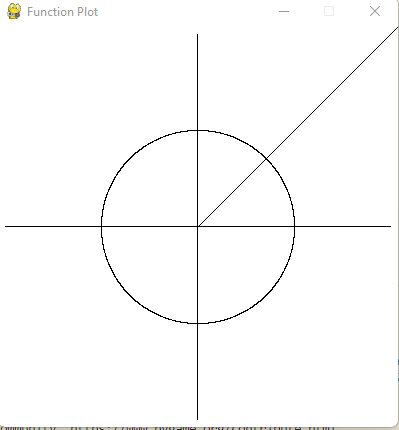
        glClear(GL\_COLOR\_BUFFER\_BIT|GL\_DEPTH\_BUFFER\_BIT)

        plotfunc()

        pygame.display.flip()

        pygame.time.wait(10)

main()



Latihan5\_3.py

import pygame

from pygame.locals import \*

from OpenGL.GL import \*

from OpenGL.GLU import \*

import numpy as np

import math

**def** plotfunc():

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    glBegin(GL\_LINES)

    glVertex2f(-2.0, 0.0)

    glVertex2f(2.0, 0.0)

    glVertex2f(0.0, 2.0)

    glVertex2f(0.0, -2.0)

    glEnd()

    for t in np.arange(0.0,6.28, 0.001):

        x = math.sin(3\*t)

        y = math.cos(5\*t)

        z = t

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glEnd()

    glFlush()

**def** main():

    pygame.init()

    display = (400,400)

    pygame.display.set\_caption('Function Plot')

    pygame.display.set\_mode(display, DOUBLEBUF|OPENGL)

    gluPerspective(45, (display[0]/display[1]), 0.1, 50.0)

    glTranslatef(0.0,0.0, -5)

    glClearColor(1.0, 1.0, 1.0, 1.0)

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit()

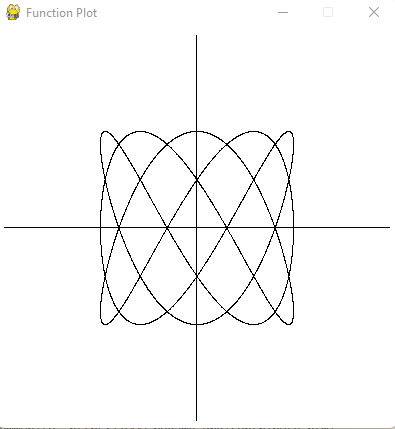
        glClear(GL\_COLOR\_BUFFER\_BIT|GL\_DEPTH\_BUFFER\_BIT)

        plotfunc()

        pygame.display.flip()

        pygame.time.wait(10)

main()



**TUGAS**

1. Tentukan persamaan garis yang baru, dengan menggeser pada sumbu y yang

menghasilkan output sbb:

import pygame

from pygame.locals import \*

from OpenGL.GL import \*

from OpenGL.GLU import \*

import numpy as np

import math

**def** plotfunc():

    glClear(GL\_COLOR\_BUFFER\_BIT)

    glColor3f(0.0, 0.0, 0.0)

    glPointSize(1.0)

    glBegin(GL\_LINES)

    glVertex2f(-2.0, 0.0)

    glVertex2f(2.0, 0.0)

    glVertex2f(0.0, 2.0)

    glVertex2f(0.0, -2.0)

    glEnd()

    for t in np.arange(-5.0,6.28, 0.001):

        x = math.sin(t)

        y = math.cos(t)

        z = t + 1

        glBegin(GL\_POINTS)

        glVertex2f(x,y)

        glVertex2f(t,z)

        glEnd()

    glFlush()

**def** main():

    pygame.init()

    display = (400,400)

    pygame.display.set\_caption('Function Plot')

    pygame.display.set\_mode(display, DOUBLEBUF|OPENGL)

    gluPerspective(45, (display[0]/display[1]), 0.1, 50.0)

    glTranslatef(0.0,0.0, -5)

    glClearColor(1.0, 1.0, 1.0, 1.0)

    while True:

        for event in pygame.event.get():

            if event.type == pygame.QUIT:

                pygame.quit()

                quit()

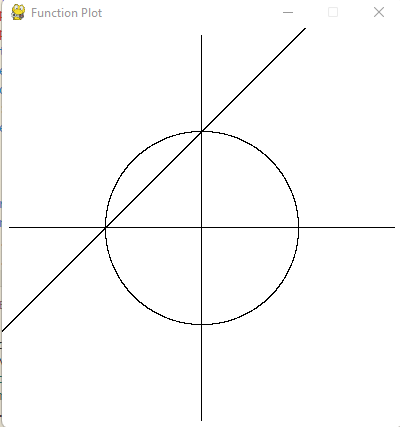
        glClear(GL\_COLOR\_BUFFER\_BIT|GL\_DEPTH\_BUFFER\_BIT)

        plotfunc()

        pygame.display.flip()

        pygame.time.wait(10)

main()



2. Kesimpulan dari latihan pada Modul 5

Dengan mengintegrasikan teori fungsi parametrik dengan penggunaan OpenGL dapat menghasilkan pemetaan grafis dari fungsi tersebut.