EXPERIMENT NO 1 **DATE: 12/08/2021**

a) LINE DRAWING: HORIZONTAL, VERTICAL AND DIAGONAL LINE

AIM

To write an option-based program that draws Horizontal, Vertical and Diagonal Lines based on User Inputs.

Option 1: Horizontal Line - ask for X-coordinate range and specific Y-coordinate and plot the line.
Option 2: Vertical Line - ask for specific X-coordinate and Y-coordinate range and plot the line.
Option 3: Diagonal Line - ask for inputs - for example: the input: 5, 10; should plot the line (5,5) (6,6) (7,7) (8,8) (9,9) (10,10).

PROGRAM

```
# Importing required modules

import OpenGL

from OpenGL.GL import *

from OpenGL.GLU import *

from OpenGL.GLUT import *


import sys


def init(): # Clear screen and set origin

    glClearColor(0.0, 0.0, 0.0, 1.0)

    gluOrtho2D(-1.0, 1.0, -1.0, 1.0)


def plot_points(points): # Function to plot the points

    glClear(GL_COLOR_BUFFER_BIT)

    glColor3f(1.0, 0.0, 0.0)

    glPointSize(10.0)

    glBegin(GL_POINTS)

    for point in points:

        glVertex2f(point[0], point[1])
```

```
    glEnd()
    glFlush()

def display_menu(): # Function to display menu
    print("-----Menu-----")
    print("1. Horizontal Line")
    print("2. Vertical Line")
    print("3. Diagonal Line")
    print("0. Exit")
    return int(input("Enter Choice:"))

def get_coordinates(choice): # Function to get coordinates
    if choice == 1:
        x1, x2 = map(int, input("Enter x-coordinate range: (Enter coordinates seperated by space. Eg. '2 4')").split(" "))
        y = int(input("Enter y coordinate: "))
        return [x1, x2], [y]
    elif choice == 2:
        x = int(input("Enter x coordinate: "))
        y1, y2 = map(int, input("Enter y-coordinate range: (Enter coordinates seperated by space. Eg. '2 4')").split(" "))
        return [x], [y1, y2]
    else:
        start, end = map(int, input("Enter start and end coordinates seperated by space. (For (1, 1) to (7,7) Enter '1 7')").split(" "))
        return [start, start], [end, end]

def diagonal_line(x, y): # Function to get points to draw diagonal line
    points = []
    while x <= y:
```

```
        points.append([x, x])
        x += 0.05 # Incrementing by small numbers to get points with less spacing
    plot_points(points)

def horizontal_line(x, y): # Function to get points to draw horizontal line
    x1, x2 = x
    points = []
    while x[0] <= x[1]:
        points.append([x[0], y])
        x[0] += 0.05
    plot_points(points)

def vertical_line(x, y): # Function to get points to draw vertical line
    points = []
    while y[0] <= y[1]:
        points.append([x, y[0]])
        y[0] += 0.05
    plot_points(points)

def display_window(choice, window_title): # Function to display window
    x, y = get_coordinates(choice)
    print("Creating Window...")
    glutInit(sys.argv)
    glutInitDisplayMode(GLUT_RGB)
    glutInitWindowSize(500, 500)
    glutInitWindowPosition(50, 50)
    glutCreateWindow(window_title)
    if len(x) + len(y) == 4: # Condition when the diagonal is chosen
        glutDisplayFunc(lambda: diagonal_line(x[0], y[0]))
```

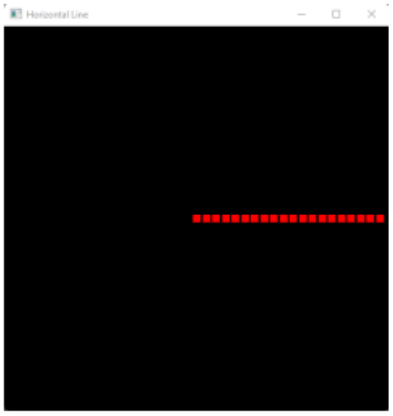
```
    elif len(x) == 2: # Condition when horizontal is chosen
        glutDisplayFunc(lambda: horizontal_line(x, y[0]))
    else: # Condition when vertical is chosen
        glutDisplayFunc(lambda: vertical_line(x[0], y))
    init()
    glutMainLoop()


def main():
    # Input dictionary for reference
    input_map = {
        1: "Horizontal Line",
        2: "Vertical Line",
        3: "Diagonal Line"
    }
    choice = 1
    while choice != 0:
        choice = display_menu()
        if choice in input_map.keys():
            # Checks if it's a valid input (i.e. present in dictionary)
            window_title = input_map[choice]
            display_window(choice, window_title)
        elif choice == 0:
            # To handle exit from program
            print("Exiting Program...")
        else:
            # To handle invalid choice
            print("Invalid Choice! Try again!")
main()
```

RESULT

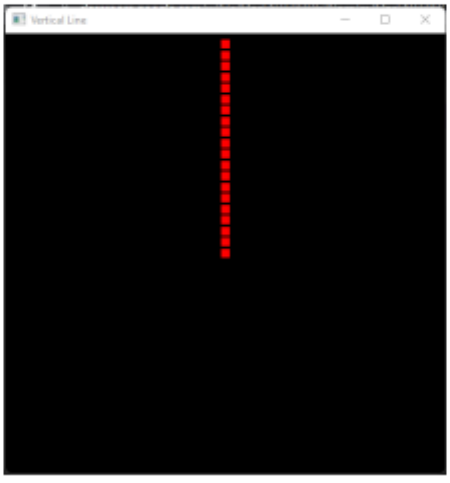
Program to draw horizontal, vertical and diagonal lines based on users' input was created and executed successfully.

INPUT/OUTPUT

```
(.venv) E:\College\S5\Computer Graphics\Experiment 1>python line_drawing.py
-----Menu-----
1. Horizontal Line
2. Vertical Line
3. Diagonal Line
0. Exit
Enter Choice:1
Enter x-coordinate range: (Enter coordinates seperated by space. Eg. '2 4')0 1
Enter y coordinate: 0
Creating Window...
```




```
(.venv) E:\College\S5\Computer Graphics\Experiment 1>python line_drawing.py
-----Menu-----
1. Horizontal Line
2. Vertical Line
3. Diagonal Line
0. Exit
Enter Choice:2
Enter x coordinate: 0
Enter y-coordinate range: (Enter coordinates seperated by space. Eg. '2 4')0 1
Creating Window...
```



```
(.venv) E:\College\S5\Computer Graphics\Experiment 1>python line_drawing.py
-----Menu-----
1. Horizontal Line
2. Vertical Line
3. Diagonal Line
0. Exit
Enter Choice:3
Enter start and end coordinates seperated by space. (For (1, 1) to (7,7) Enter '1 7')0 1
Creating Window...
```