

Lab No 2

IMPLEMENTATION OF DDA LINE DRAWING ALGORITHM (PYTHON)

Computer Graphics

Instructor: Sushant Pandey

Report Submitted on: **2082/08/28**



Department of Electronics & Computer Engineering

Himalaya College of Engineering, TU

Chyasal, Lalitpur

Objective(s):

- i) To understand the Digital Differential Analyzer (DDA) line drawing algorithm.
- ii) To implement the DDA algorithm in Python and visualize the generated pixels.
- iii) To compare DDA with the analytical line equation.

Software(s) Required

Python 3, matplotlib, any IDE or Jupyter Notebook.

Theory

In raster graphics, a straight line must be approximated using discrete pixels. The DDA algorithm is an incremental method that uses the line slope to step along the dominant axis and compute intermediate points.

Given two endpoints (x_1, y_1) and (x_2, y_2) , we define:

$$d_x = x_2 - x_1, d_y = y_2 - y_1$$

The number of steps is chosen as:

$$\text{steps} = \max(|dx|, |dy|)$$

The increment in each step is:

$$x_{\text{inc}} = dx/\text{steps}$$

$$y_{\text{inc}} = dy/\text{steps}$$

Starting from (x_1, y_1) , the algorithm adds these increments repeatedly and rounds to the nearest pixel.

Algorithm: DDA Line Drawing

1. Read starting point (x_1, y_1) and ending point (x_2, y_2) .
2. Compute dx , dy and number of steps.
3. Compute x_{inc} and y_{inc} .
4. Initialise $x = x_1$, $y = y_1$.
5. For $k = 0$ to steps:
 - Plot pixel at $(\text{round}(x), \text{round}(y))$.
 - Update $x = x + x_{\text{inc}}$, $y = y + y_{\text{inc}}$.

Lab Assignment

1. Extend the DDA program to draw a rectangle given two opposite corners.

The screenshot shows a code editor interface with two terminal windows. The top terminal window displays the code for `lab2.py`, which contains a function `dda_line` for drawing a line using the DDA algorithm and a main block for drawing a rectangle. The bottom terminal window shows the execution of the script and its output, including user input for rectangle vertices and the resulting plot.

```
C:\> Users > MSI > Desktop > python > lab2.py > dda_line
1 import matplotlib.pyplot as plt
2
3 def dda_line(x1, y1, x2, y2, px, py):
4     dx = x2 - x1
5     dy = y2 - y1
6     steps = int(max(abs(dx), abs(dy)))
7     x_inc = dx / steps
8     y_inc = dy / steps
9     x, y = x1, y1
10    for _ in range(steps + 1):
11        px.append(round(x))
12        py.append(round(y))
13        x += x_inc
14        y += y_inc
15
16    x1 = int(input("Enter x1: "))
17    y1 = int(input("Enter y1: "))
18    x2 = int(input("Enter x2: "))
19    y2 = int(input("Enter y2: "))
20
21    x3, y3 = x1, y2
22    x4, y4 = x2, y1
23
24    px, py = [], []
25
26    dda_line(x1, y1, x3, y3, px, py)
27    dda_line(x3, y3, x2, y2, px, py)
28    dda_line(x2, y2, x4, y4, px, py)
29    dda_line(x4, y4, x1, y1, px, py)
30
31    plt.plot(px, py, linestyle="-", color='blue')
32    plt.title("Rectangle using DDA Algorithm")
33    plt.xlabel("X-axis")
34    plt.ylabel("Y-axis")
35    plt.gca().set_aspect("equal", adjustable="box")
36    plt.grid(True)
37    plt.show()
38

PS C:\Users\MSI\Desktop\coding> & "C:/Program Files/Python312/python.exe" c:/Users/MSI/Desktop/python/lab2.py
Enter x1:
Enter y1:
Enter x2:
Enter y2:
Ln 10, Col 31  Spaces: 4  UTF-8  CRLF  { } Python  3.12.5  🔍
```

```
25
26    dda_line(x1, y1, x3, y3, px, py)
27    dda_line(x3, y3, x2, y2, px, py)
28    dda_line(x2, y2, x4, y4, px, py)
29    dda_line(x4, y4, x1, y1, px, py)
30
31    plt.plot(px, py, linestyle="-", color='blue')
32    plt.title("Rectangle using DDA Algorithm")
33    plt.xlabel("X-axis")
34    plt.ylabel("Y-axis")
35    plt.gca().set_aspect("equal", adjustable="box")
36    plt.grid(True)
37    plt.show()
38

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  +  🔍  powershell
PS C:\Users\MSI\Desktop\coding> & "C:/Program Files/Python312/python.exe" c:/Users/MSI/Desktop/python/lab2.py
Enter x1: 0
Enter y1: 0
Enter x2: 10
Enter y2: 10
Ln 10, Col 31  Spaces: 4  UTF-8  CRLF  { } Python  3.12.5  🔍
```

The screenshot shows a Python code editor with a terminal window below it. The code in `lab2.py` uses the DDA algorithm to draw a rectangle on a grid. The terminal shows user input for coordinates and the resulting plot titled "Rectangle using DDA Algorithm".

```

17 y1 = int(input("Enter y1: "))
18 x2 = int(input("Enter x2: "))
19 y2 = int(input("Enter y2: "))
20
21 x3, y3
22 x4, y4
23
24 px, py
25
26 dda_li
27 dda_li
28 dda_li
29 dda_li
30
31 plt.pl
32 plt.ti
33 plt.xl
34 plt.yl
35 plt.gc
36 plt.gr
37 plt.sh
38

```

PROBLEMS OUTPUT

PS C:\Users\MSI> Enter x1: 0
Enter y1: 0
Enter x2: 10
Enter y2: 10

Figure 1

Rectangle using DDA Algorithm

X-axis Y-axis

0 2 4 6 8 10

0 2 4 6 8 10

Ln 10, Col 31 Spaces:4 UTF-8 CRLF { } Python 3.12.5

2. Use DDA to draw the axes of a simple coordinate system (X and Y axes).

The screenshot shows a Python code editor with a terminal window below it. The code in `lab2.2.py` uses the DDA algorithm to draw the X and Y axes of a coordinate system. The terminal shows user input for axis endpoints and the resulting plot.

```

1 import matplotlib.pyplot as plt
2
3 def dda_line(x1, y1, x2, y2):
4     dx = x2 - x1
5     dy = y2 - y1
6     steps = int(max(abs(dx), abs(dy)))
7     x_inc = dx / steps
8     y_inc = dy / steps
9     x, y = x1, y1
10    px, py = [], []
11    for _ in range(steps + 1):
12        px.append(round(x))
13        py.append(round(y))
14        x += x_inc
15        y += y_inc
16    return px, py
17
18 x_min, x_max = -10, 10
19 y_min, y_max = -10, 10
20
21 px_x, py_x = dda_line(x_min, 0, x_max, 0)
22 px_y, py_y = dda_line(0, y_min, 0, y_max)
23
24 plt.plot(px_x, py_x, color='black') # X-axis
25 plt.plot(px_y, py_y, color='black') # Y-axis
26

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\MSI\Desktop\coding> & "C:/Program Files/Python312/python.exe" c:/Users/MSI/Desktop/python/lab2.py

Enter x1: 0
Enter y1: 0
Enter x2: 10
Enter y2: 10

PS C:\Users\MSI\Desktop\coding> []

Ln 6, Col 39 Spaces:4 UTF-8 CRLF { } Python 3.12.5

```
26 plt.title("Simple Coordinate System using DDA")
27 plt.xlabel("X-axis")
28 plt.ylabel("Y-axis")
29 plt.gca().set_aspect("equal", adjustable="box")
30 plt.grid(True)
31 plt.show()
32
```

TERMINAL

```
PS C:\Users\MSI\Desktop\coding> & "C:/Program Files/Python312/python.exe" c:/Users/MSI/Desktop/python/lab2.2.py
```

Figure 1

Simple Coordinate System using DDA

(x, y) = (8.04, -7.42)

Conclusion

In this lab we gained the practical experience in handling geometric computations using matplotlib. We learned the concept of DDA line drawing algorithm and made the rectangle using it.