Batch: C2 Roll No.: 16010121110

Experiment No. 02

TITLE: Drawing of line using computer graphics.

AIM:
Generate the line using computer graphics program

Use following links to perform on Virtual Lab

https://cse18-iiith.vlabs.ac.in/exp/coordinate-systems/pretest.html

https://github.com/ptp28/cg-vlab

Expected OUTCOME of Experiment:

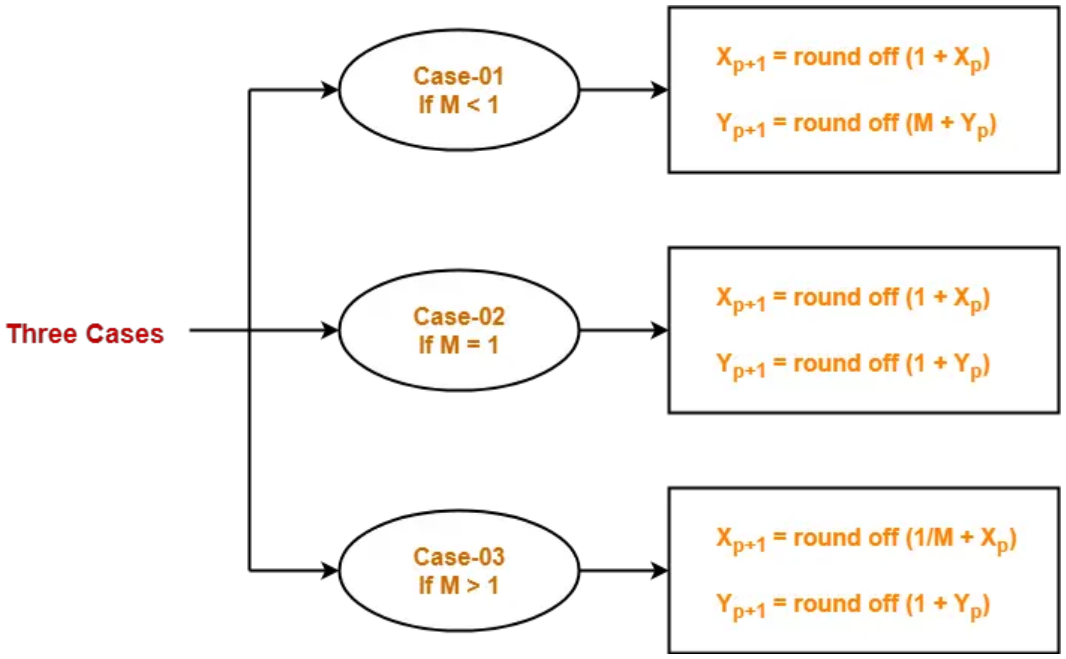
Understand various line drawing algorithms in computer graphics

---

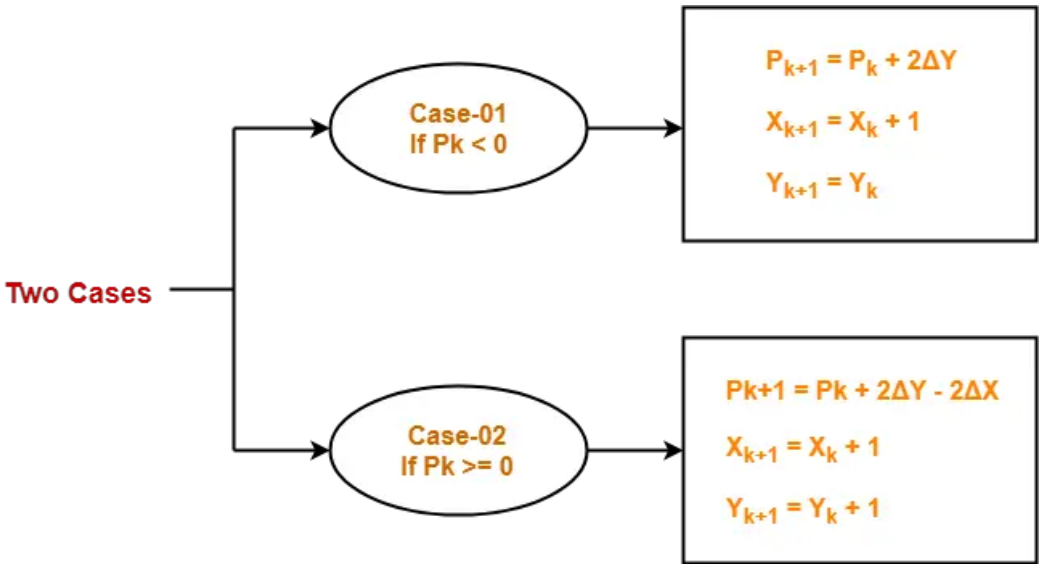
Books/ Journals/ Websites referred:

W3Schools
Geeks for Geeks

Algorithm: a



Algorithm: b



Implementation details:

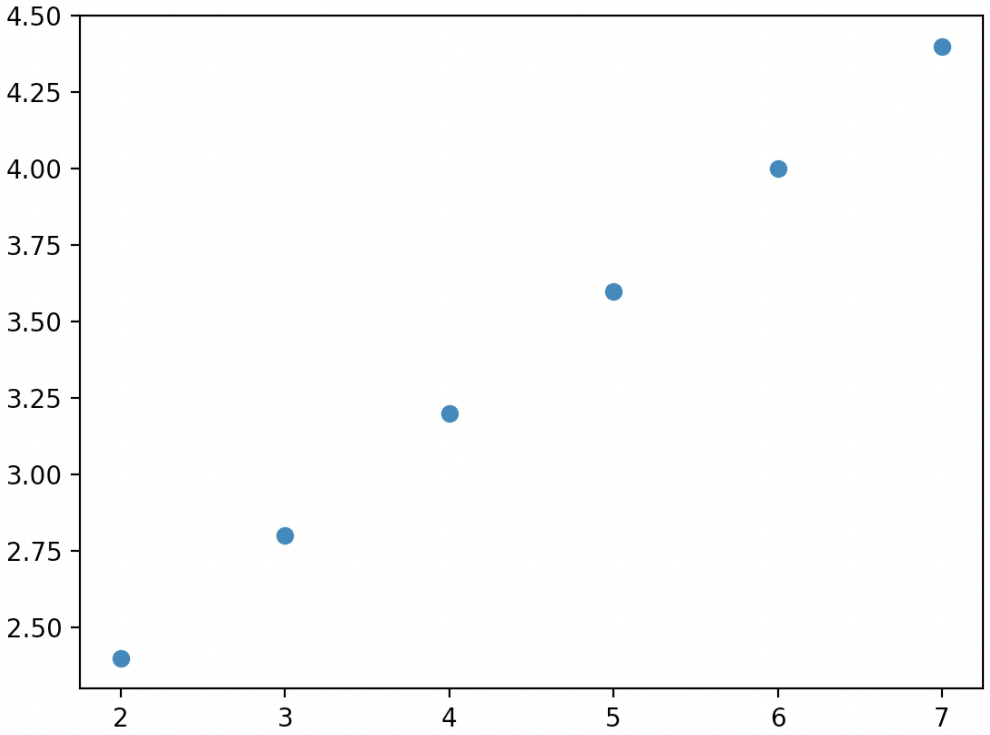
```
import matplotlib.pyplot as plt
def DDA(x1,y1,x2,y2):
```

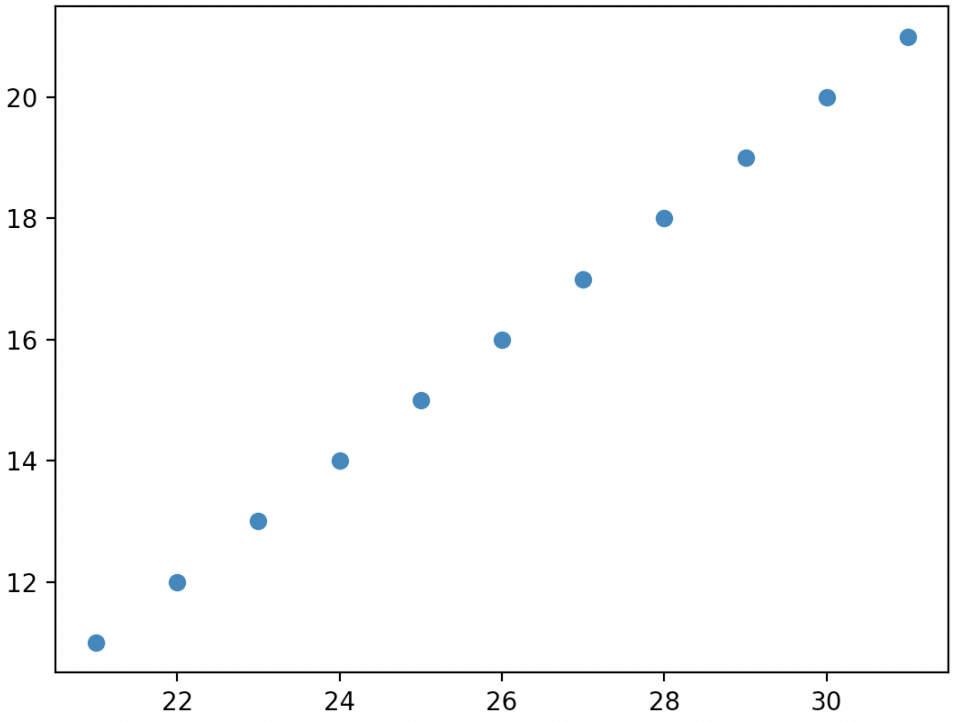
```
dy=y2-y1
dx=x2-x1
m=dy/dx
x=x1
y=y1
X=[]
Y=[]
while(x<=x2 or y<=y2):
if(m<1):
x=x1+1
y=y1+m
if(m>1):
y=y1+1
x=x1+1/m
if(m==1):
x=x1+1
y=y1+1
print(x,y)
X.append(x)
Y.append(y)
x1=x
y1=y
plt.scatter(X,Y)
plt.show()
DDA(1,2,6,4)
```

```
from OpenGL.GL import *
from OpenGL.GLUT import *
from OpenGL.GLU import *
```

```
def draw_line(x1, y1, x2, y2):
glVertex2f(x1, y1)
glVertex2f(x2, y2)

def display():
glClear(GL_COLOR_BUFFER_BIT)
glColor3f(1.0, 1.0, 1.0) # Set color to white (RGB
values: 1.0, 1.0, 1.0)
glBegin(GL_LINES)

draw_line(0.0, 0.0, 0.0, 0.5)
draw_line(0.0, 0.5, 0.5, 0.5)
draw_line(0.5, 0.5, 0.5, 0.0)
draw_line(0.5, 0.0, 0.0, 0.0)

glEnd()
glFlush()



glutInit()
glutInitWindowSize(400, 400)
glutCreateWindow(b"PyOpenGL Line Example")
glutDisplayFunc(display)
glutMainLoop()
```

Output(s) (final edited screen shot):


4.50
4.25
4.00
3.75
3.50
3.25
3.00
2.75
2.50
2
3
4
5
6
7


20
18
16
14
12
22
24
26
28
30

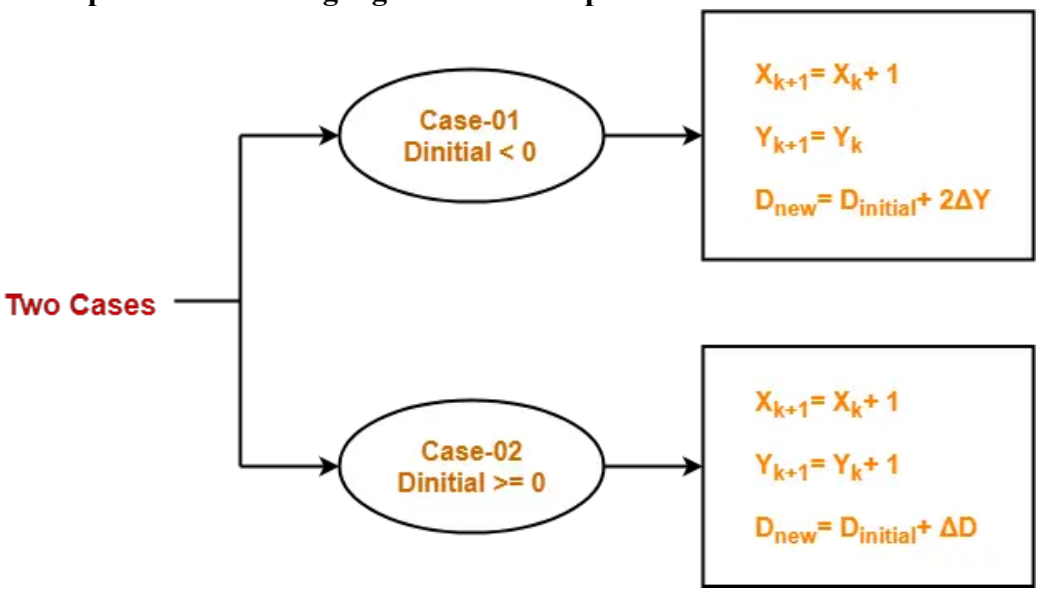
Conclusion and discussion (Comparative - compulsory):
We have understood two line drawing algorithms that are used in computer graphics. One of them is the Bresenham line drawing algorithm while the other is the Digital differential analyzer algorithm (DDA). Bresenham's Algorithm is faster than DDA Algorithm in line because it involves only addition & subtraction in its calculation and uses only integer arithmetic.

Screenshots from Vlab (Compulsory)

Date:

Signature of faculty in-charge

Explain Mid-point line drawing algorithm and implement it




```
import matplotlib.pyplot as plt
def MLD(x1,y1,x2,y2):
dy=y2-y1
dx=x2-x1
d=2*dy-dx
dd =2*dy-2*dx
x=x1
y=y1
```

```
X=[]
Y=[]
while(x<=x2 or y<=y2):
if(d<0):
x=x1+1
y=y1
d=d+2*dy
if(d>=0):
y=y1+1
x=x1+1
d=d+dd
print(x,y,d)
X.append(x)
Y.append(y)
x1=x
y1=y
plt.scatter(X,Y)
plt.show()
MLD(20,10,30,18)
```