Computer Graphics

Homework 1: DDA Algorithm

From class slide, the code is in C language

```
#include "device.h"
#define ROUND(a) ((int)(a+0.5))
void lineDDA (int xa, int ya, int xb, int yb)
{
        int dx = xb – xa, dy = yb – ya, steps, k;
        float xIncrement, yIncrement, x = xa, y = ya;

        if(abs(dx)>abs(dy)) steps = abs(dx);
        else steps = abs(dy);
        xIncrement = dx/(float)steps;
        yIncrement = dy/(float)steps;

        setPixel(ROUND(x), ROUND(y));
        for(k=0; k<steps; k++){
                x += xIncrement;
                y += yIncrement;
                setPixel(ROUND(x), ROUND(y));
        }
}
```

Code implementation in Python

Colab notebook: [DDA Algorithm]

```
def lineDDA(xa,ya,xb,yb):

  def ROUND(a):
    return int(a + 0.5)

  def setPixel(x, y):
    xs, ys = [], []
    for k in range(steps):
      x += xIncrement
      y += yIncrement
      xs.append(x); ys.append(y)
      print('x = %s, y = %s' % (ROUND(x),ROUND(y)))
    return xs, ys

  x, y = xa, ya
  dx, dy = xb-xa, yb-ya
  steps = abs(dx) if abs(dx) > abs(dy) else abs(dy)
  xIncrement = dx/float(steps)
  yIncrement = dy/float(steps)
  print ('x = %s, y = %s' % (ROUND(x),ROUND(y)))

  # Loop and store the x, y points
  xs, ys = setPixel(ROUND(x), ROUND(y))

  # Display the line
  plt.plot(xs, ys)
  plt.show()

lineDDA(2,5,10,20)
```

Output:

```
x = 2, y = 5
x = 3, y = 6
x = 3, y = 7
x = 4, y = 8
x = 4, y = 9
x = 5, y = 10
x = 5, y = 11
x = 6, y = 12
x = 6, y = 13
x = 7, y = 14
x = 7, y = 15
x = 8, y = 16
x = 8, y = 17
x = 9, y = 18
x = 9, y = 19
x = 10, y = 20
```