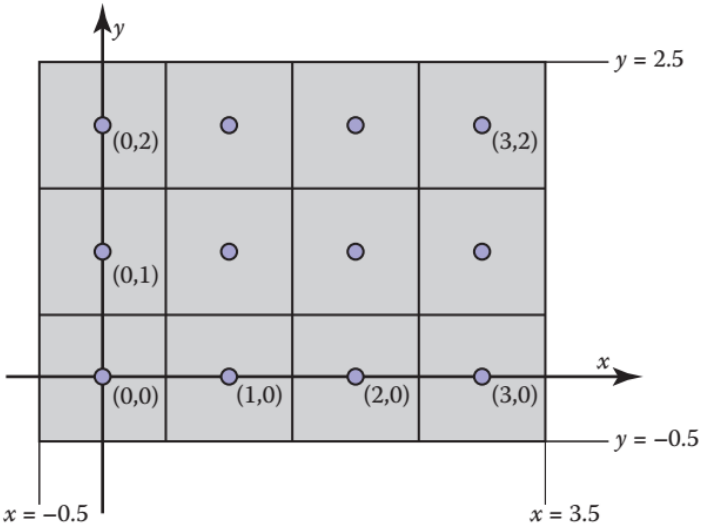
CS 418: Interactive Computer Graphics Spring 2019
In-class Worksheet

Drawing Lines

Imagine you have managed to time-travel back to the 1960s and have secured a job as software engineer developing a graphics library. Your first assignment is to devise an algorithm to draw a line from a start pixel (x_0, y_0) to an end pixel (x_1, y_1). This means you need to decide which pixels to light up between the start and end pixel.

Think of the grid of pixels as they are shown in this diagram:



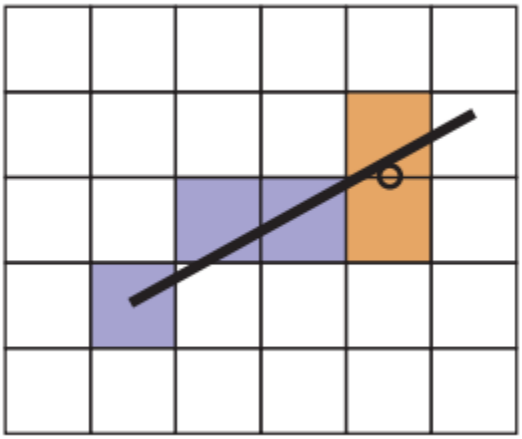
You can assume the start and end pixel are located at integer coordinates and that you can compute an equation for the line y=mx+b from the endpoints. You notice that the line is not guaranteed go exactly through integer-valued (x,y) coordinates.

That's too hard…but I could solve a simpler problem

After thinking about it, you decide it would be a lot easier to rasterize a line that starts at (0,0) and has a slope m<1. How could you transform the endpoints of the line to achieve that? Consider using two steps…first translating and then reflection.

Light it up

Now, devise an algorithm that lights up pixels between (0,0) and (x_{end}, y_{end}). Consider the following picture as a hint…the algorithm is trying to decide which of the orange pixels to light up to rasterize the black line.



Some questions to consider:

1. The algorithm should be iterative…what are you looping over?
2. If you just lit pixel (x,y) what are the candidates for the next pixel to light?
3. How can you use y=mx+b to determine which pixel to light up?
   Consider the implicit form f(x,y)=mx+b-y