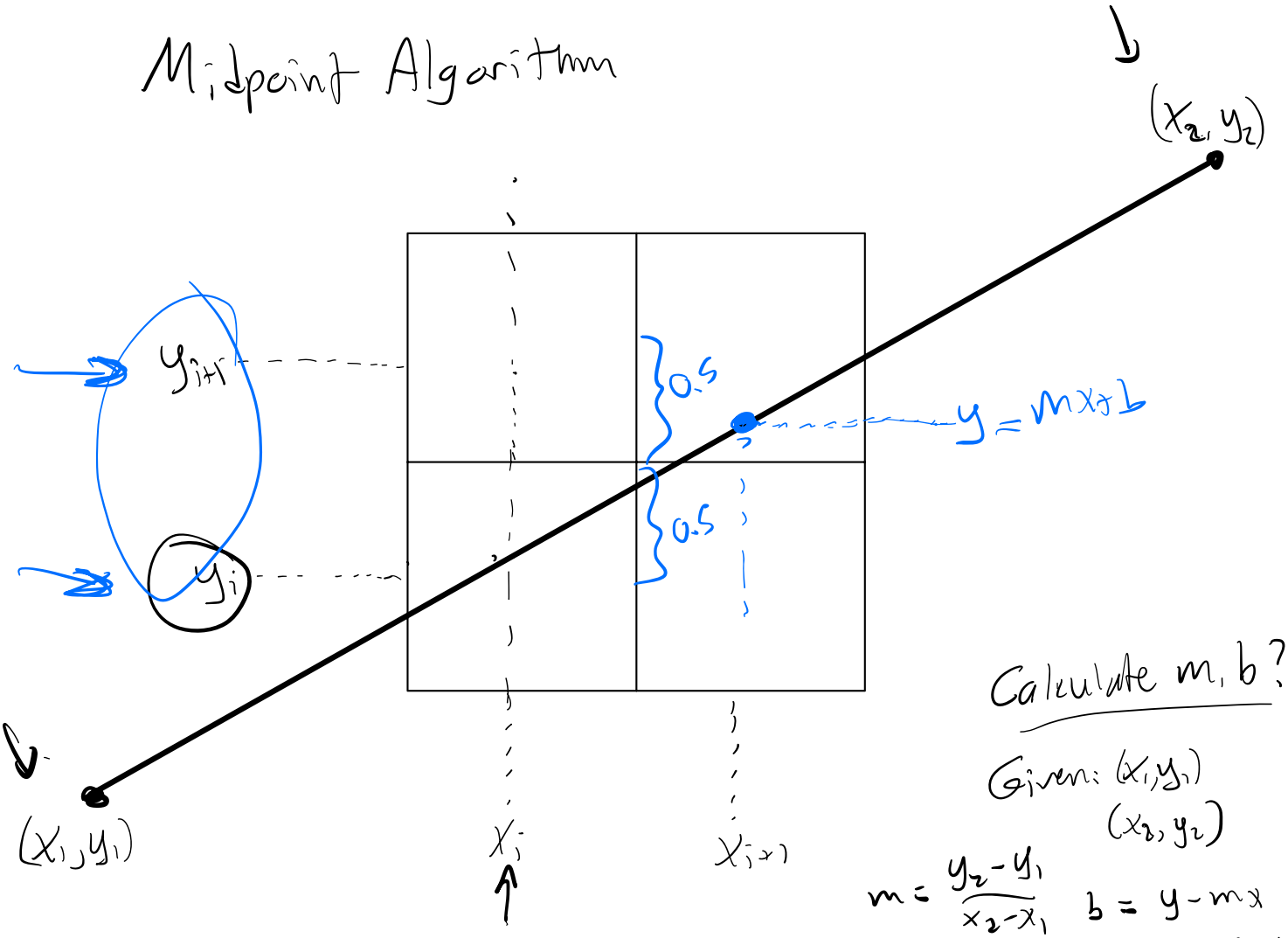
Midpoint Algorithm



Calculate m, b?

Given: (x_1, y_1)
(x_2, y_2)

$m = \frac{y_2 - y_1}{x_2 - x_1}$ $b = y - mx$
$b = y_1 - m x_1$

```
slow_line(p1, p2):
    // compute m, b
    for x = ceil(x1) : floor(x2):
        y = b + mx
        draw( x , round(y))
```

x y

Efficiency?

1 add
1 mult
1 round

```
fast_line(p1, p2):
    // compute m, b
    y = b + m xmin
    for x = ceil(x1) : floor(x2):
        y += m
        draw(x, round(y))
```

Efficiency?

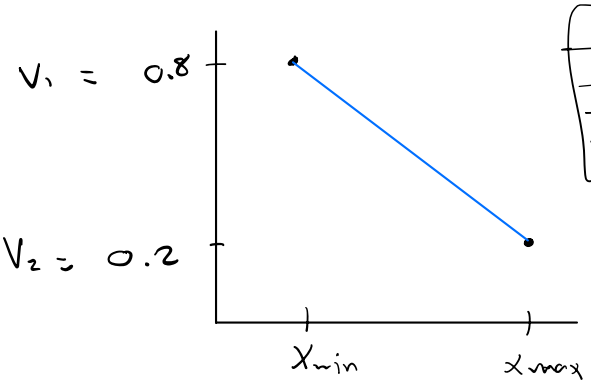
Interpolating Values

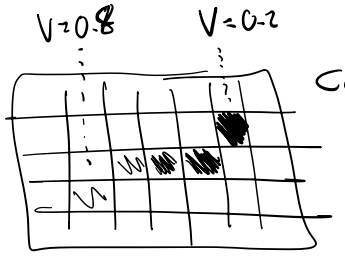
Given $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ and V_1, V_2,

Scalar values of some attribute at the endpoints,

e.g. color

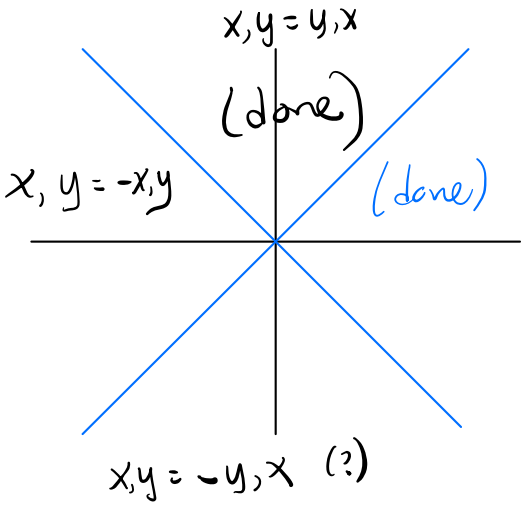
interpolate values at each pixel drawn.






```
Colored_line(P1, P2, V1, V2):
    // calc m, b
    // calc mv bv
    for x = xmin : xmax:
        y = mx+b
        v = mv x + bv
        draw(x, round(y), v)
```

Other ~~Quadrants~~ octants?



Parametric view:

$$V(x) = V_1 + \overset{x}{(x - x_1)} \overset{m}{\frac{(V_2 - V_1)}{(x_2 - x)}}$$

$$V_1 + t\,(V_2 - V_1)$$

$$V(t) = (1-t)V_1 + t(V_2)$$