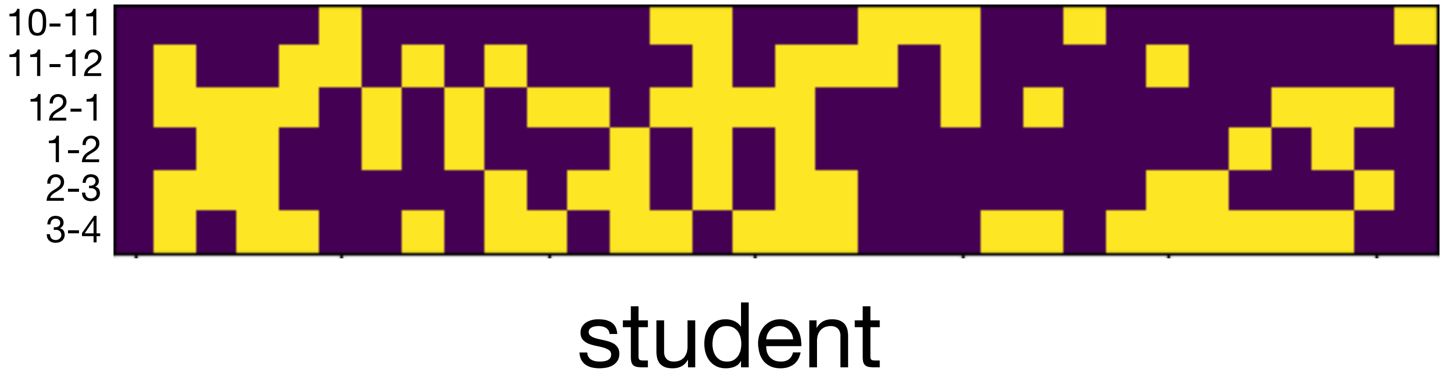
Computer Graphics

Lecture 2

Julia; Vectors; (Modeling)

Announcements



- Office hours adjusted slightly:
  - M 10-11, W 11-12, F 11:30-12:30, and by appointment
- HW0 due Friday!
  - Some math with vectors, to achieve point-in-triangle test
- A0 due Monday!
  - Draw a triangle! Note: code due Monday, artifact due Tuesday
- Likely: there will be videos to watch for Wednesday

Goals

- Know how to manipulate the pixel values of an image in Julia
- Understand that raster images are *sampled* from an underlying ideal image.
- Be able to work with and interpret basic vector operations, including addition, subtraction, dot, and cross products.

Roadmap

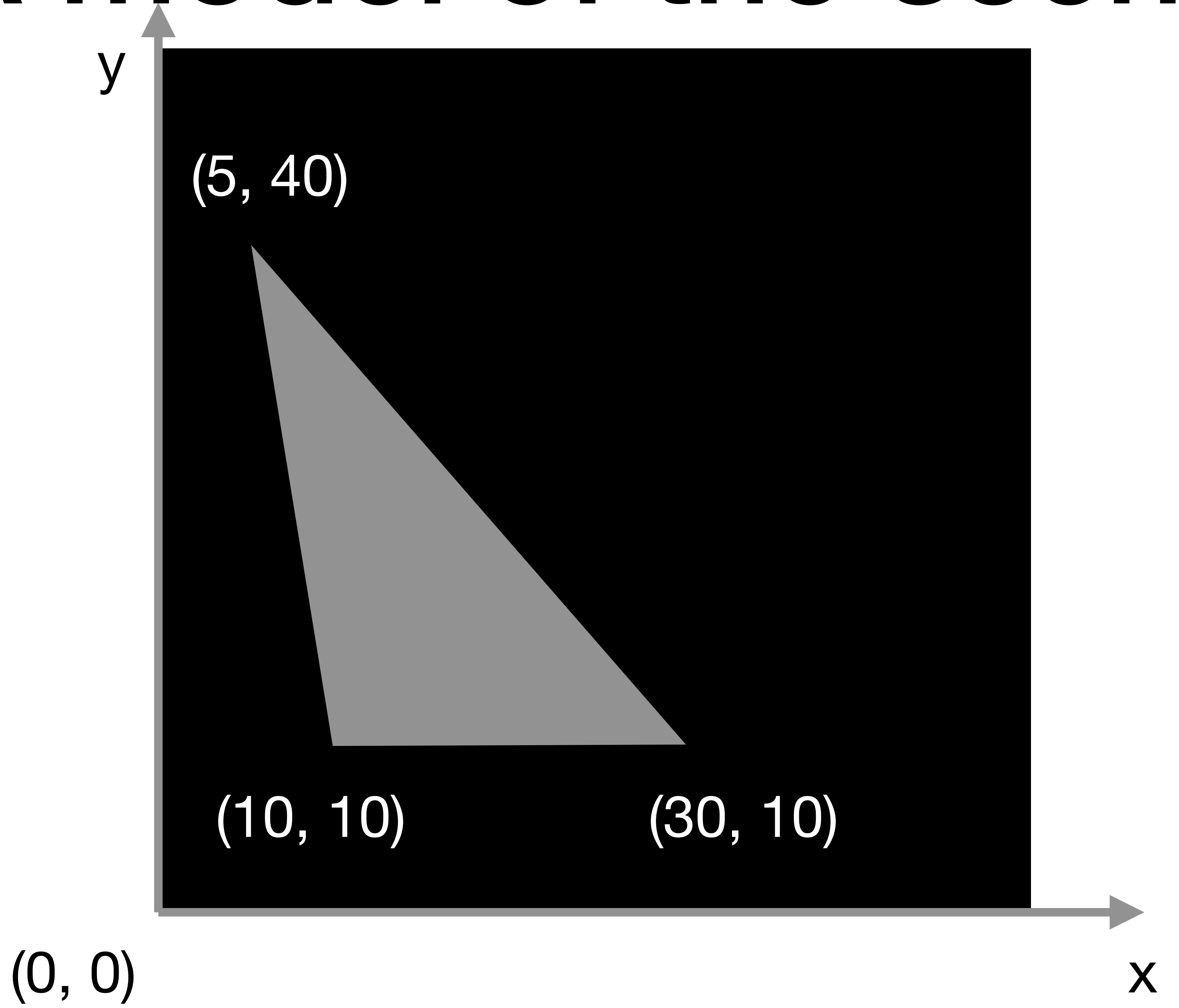
- Last time: The Simplest Possible Graphics System(TM) that is A0.

    1. Model a triangle (3 vertex positions, counter-clockwise)
    2. Draw an image of-- wait, what's an image?

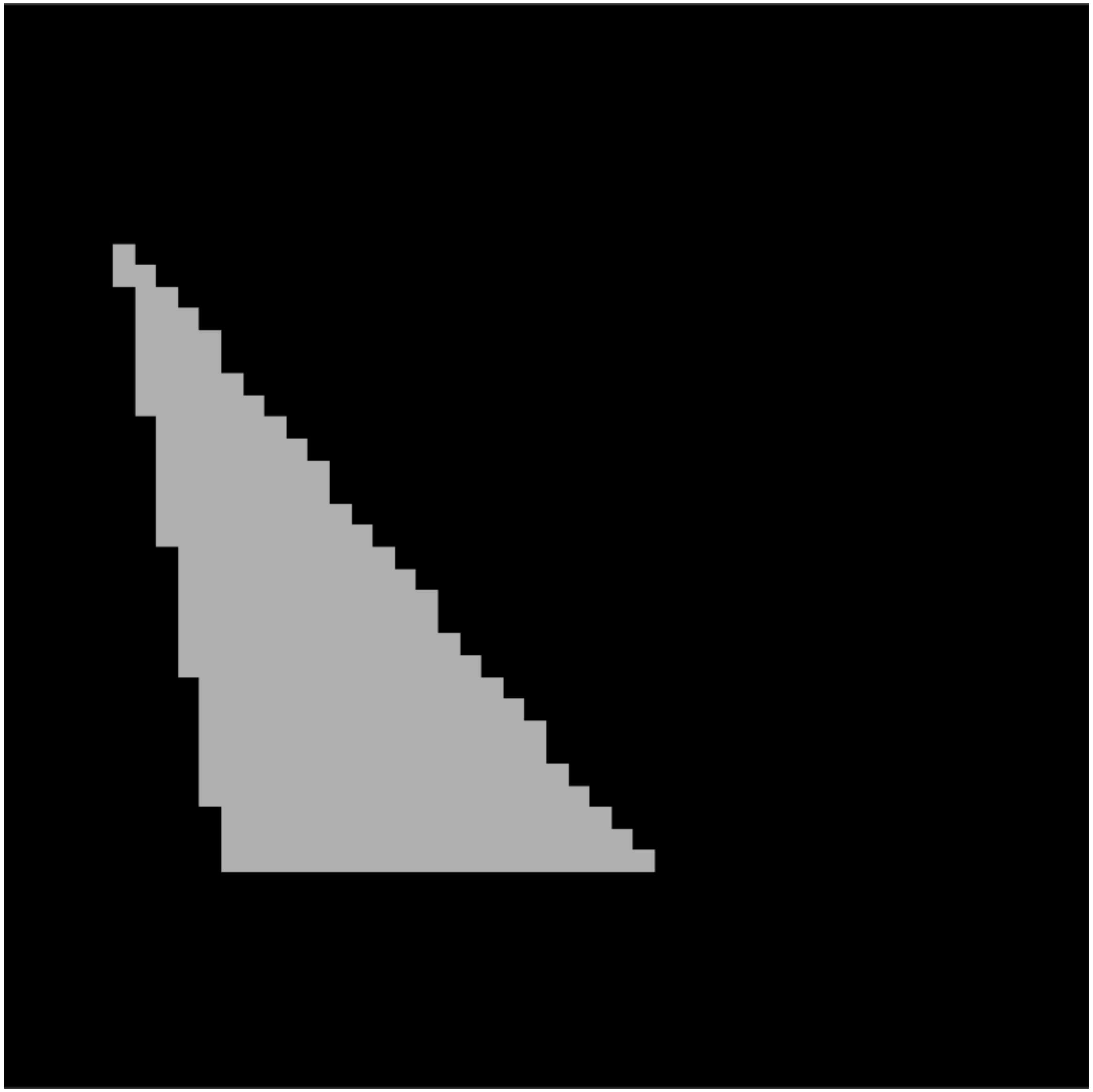
- Today: finishing up the Simplest Possible Graphics System(TM)

    3. Draw an image of the triangle

A model of the scene



A Raster Image of the Scene

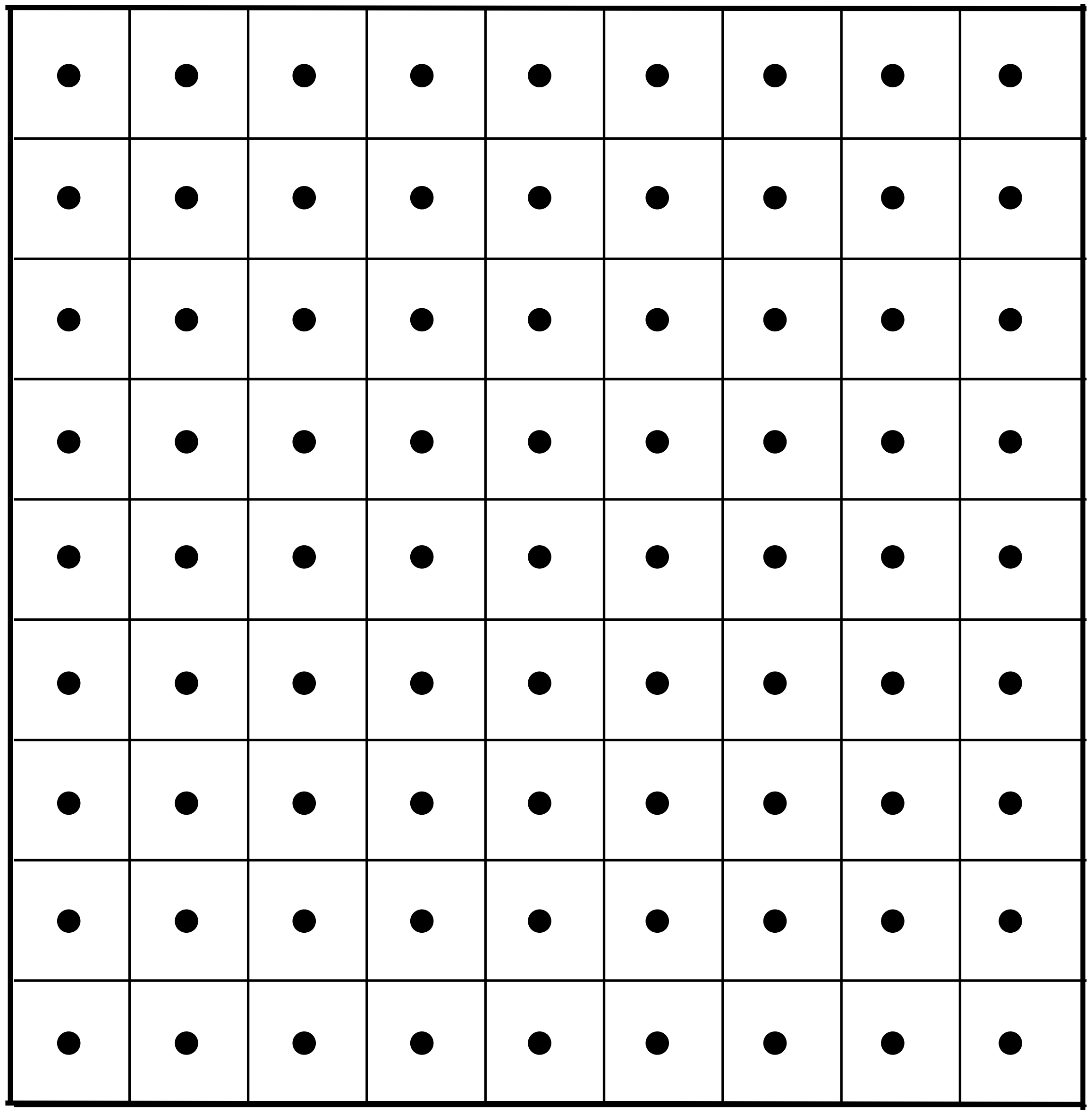


Raster images are *sampled*

function that maps 2D *positions* to *distributions of radiant energy*

Representing Raster Images

What do pixels *mean*?



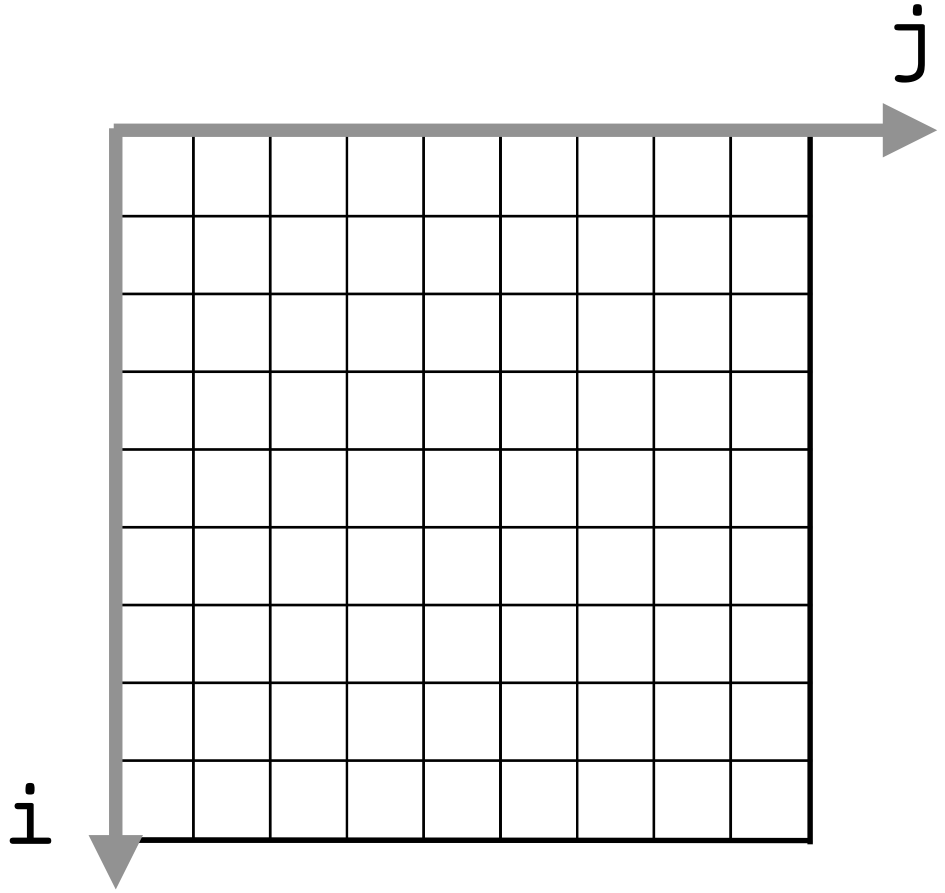
Convention: a pixel gets the color sampled at the ***center*** of the pixel.

2D Arrays in Julia

```
canvas = zeros(RGB{Float32}, height, width)
```

Matrix-style **1-based** indexing (row, column):

```
canvas[i, j] # is the i'th row, j'th column
```



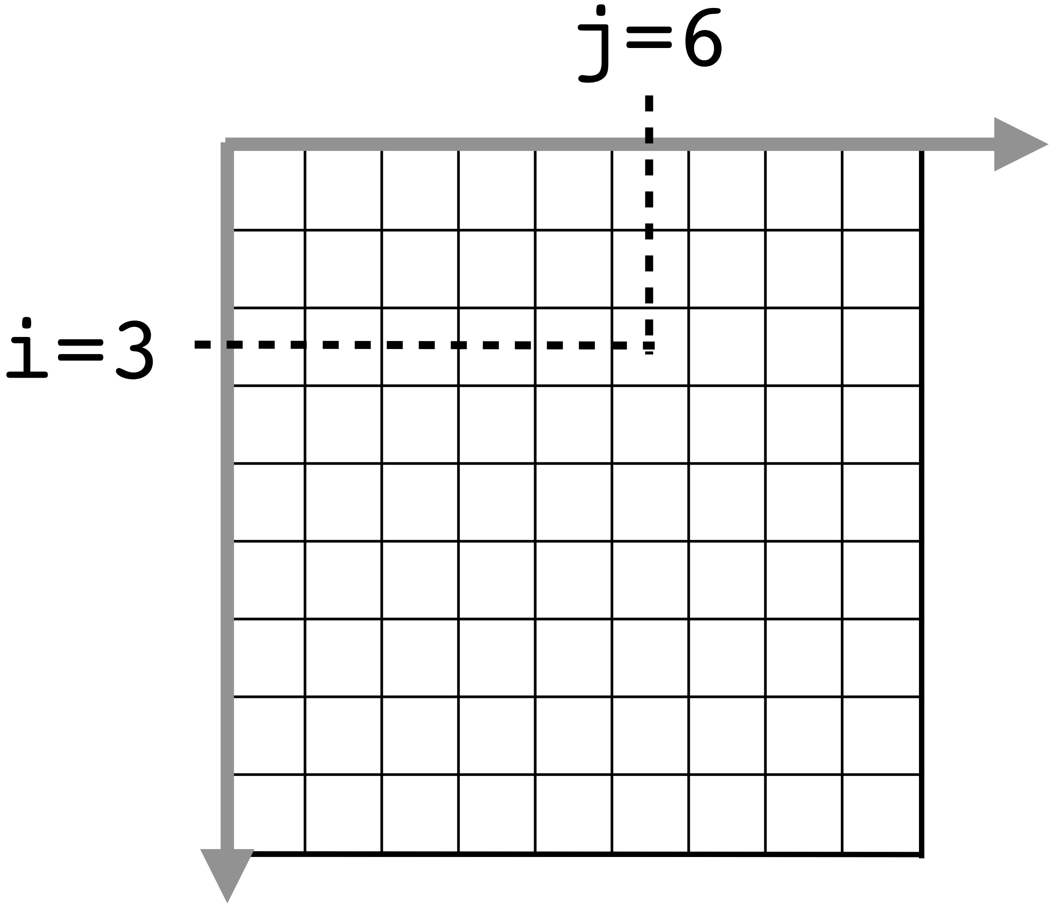
2D Arrays in Julia

```
canvas = zeros(RGB{Float32}, height, width)
```

Matrix-style **1-based** indexing (row, column):

```
canvas[i, j] # is the i'th row, j'th column
```

```
canvas[3, 6]
```



2D Arrays in Julia

```
canvas = zeros(RGB{Float32}, height, width)
```

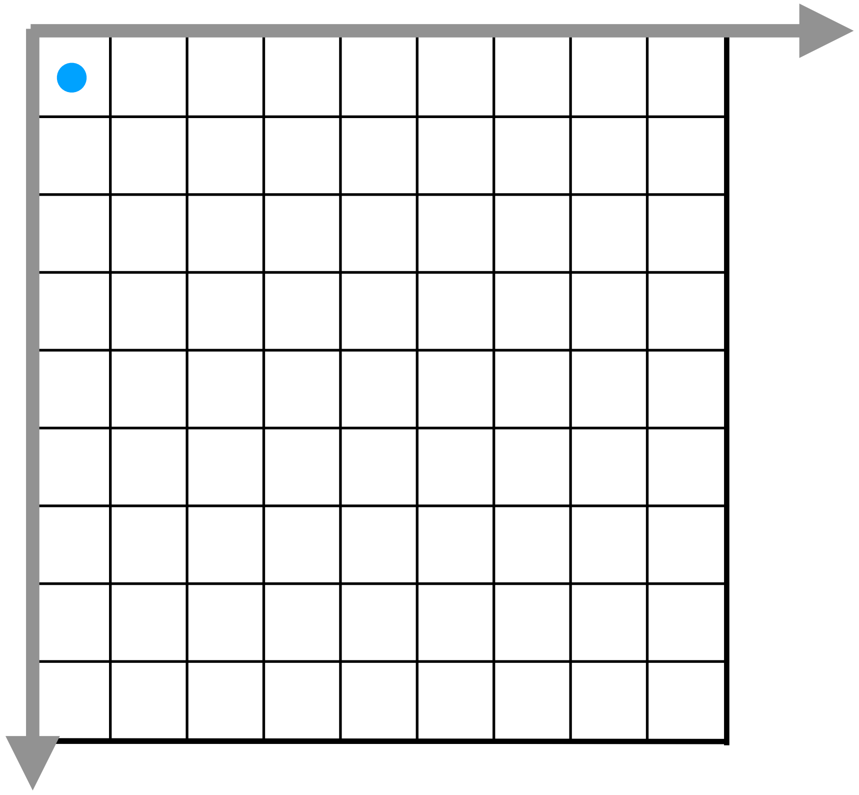
Matrix-style **1-based** indexing (row, column):

```
canvas[i, j] # is the i'th row, j'th column
```

A B
C D

What are the pixel coordinates of the blue point (the center of the top-left pixel)?

A. (0.5, 0.5)
B. (1, 1)
C. (0, 9)
D. (1, 9)



2D Arrays in Julia

```
canvas = zeros(RGB{Float32}, height, width)
```

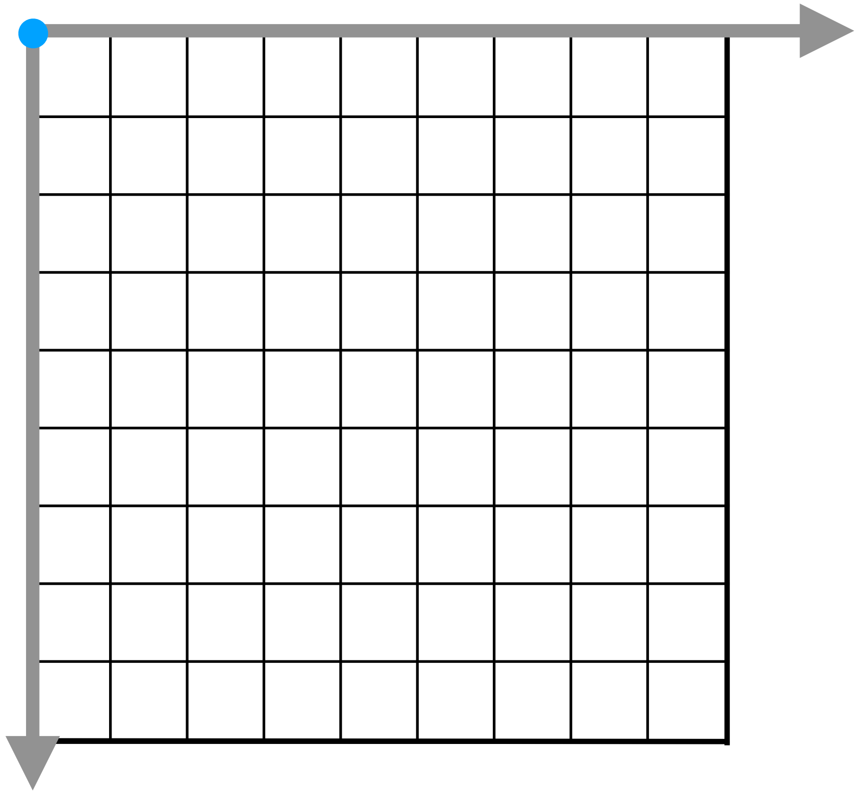
Matrix-style **1-based** indexing (row, column):

```
canvas[i, j] # is the i'th row, j'th column
```

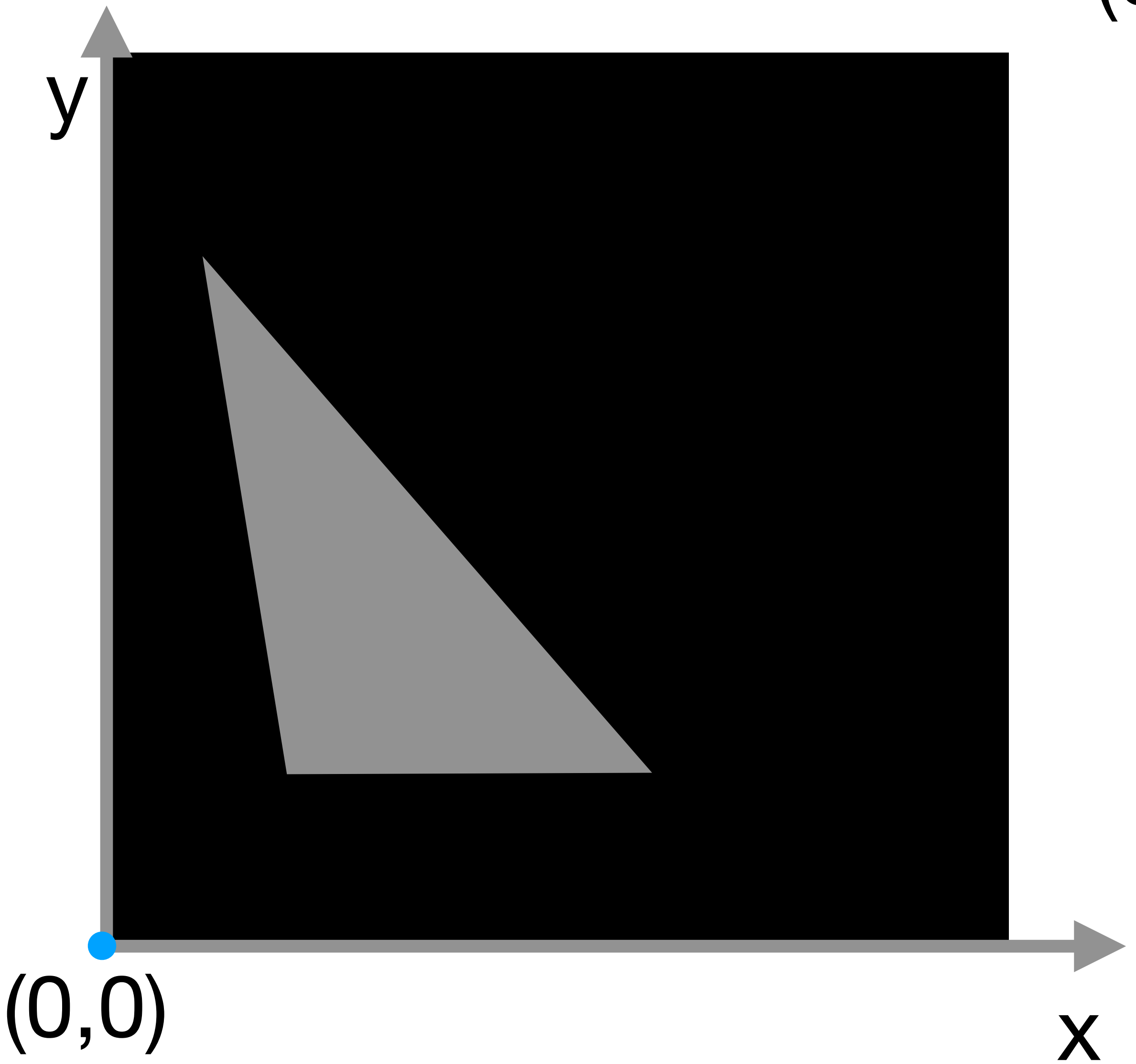
A B
C D

What are the pixel coordinates of the top-left corner of the top left pixel?

A. (0.5, 0.5)
B. (1, 1)
C. (0, 9)
D. (1, 9)



Raster Images: Coordinate Systems



Model coordinates

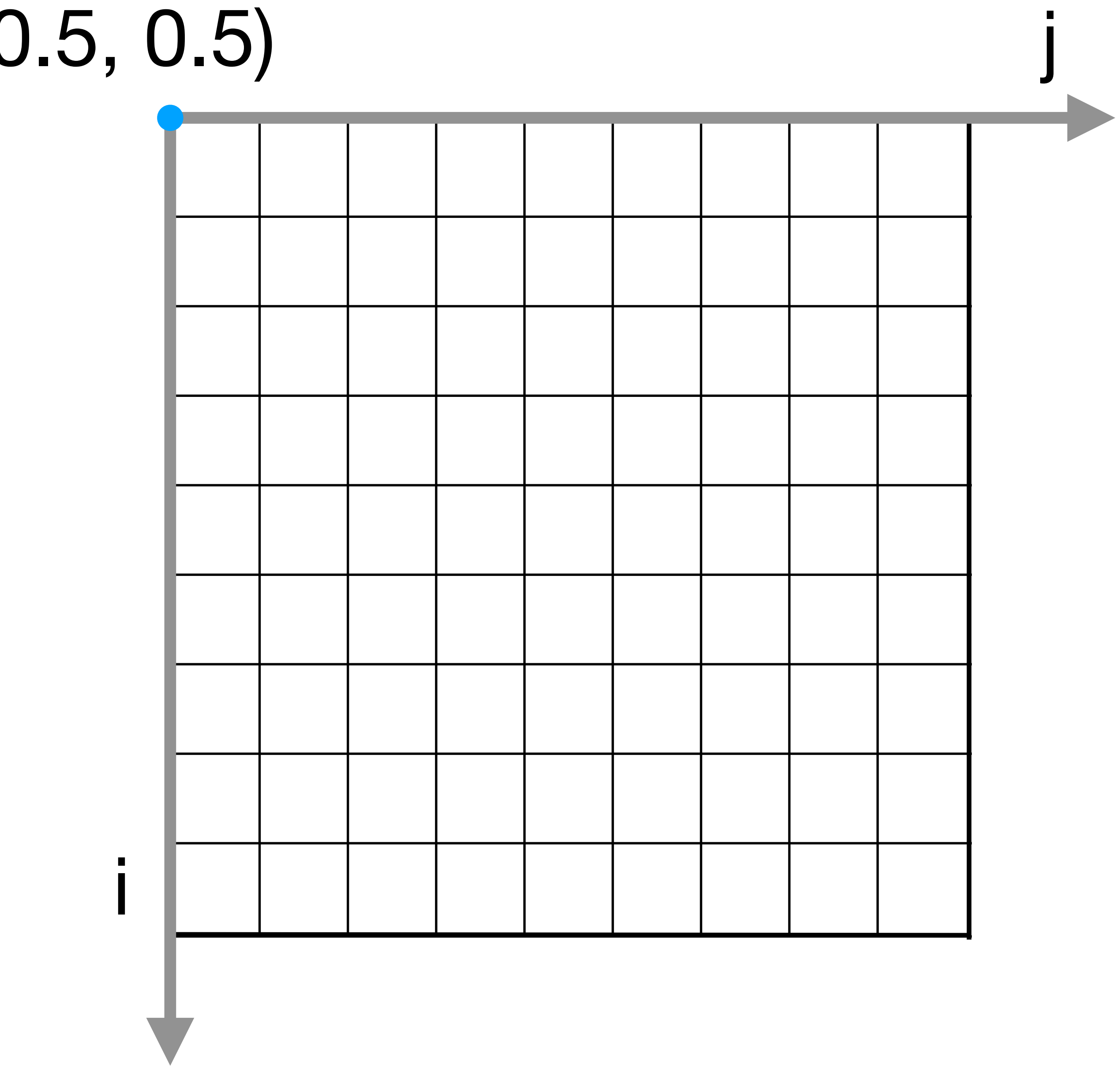
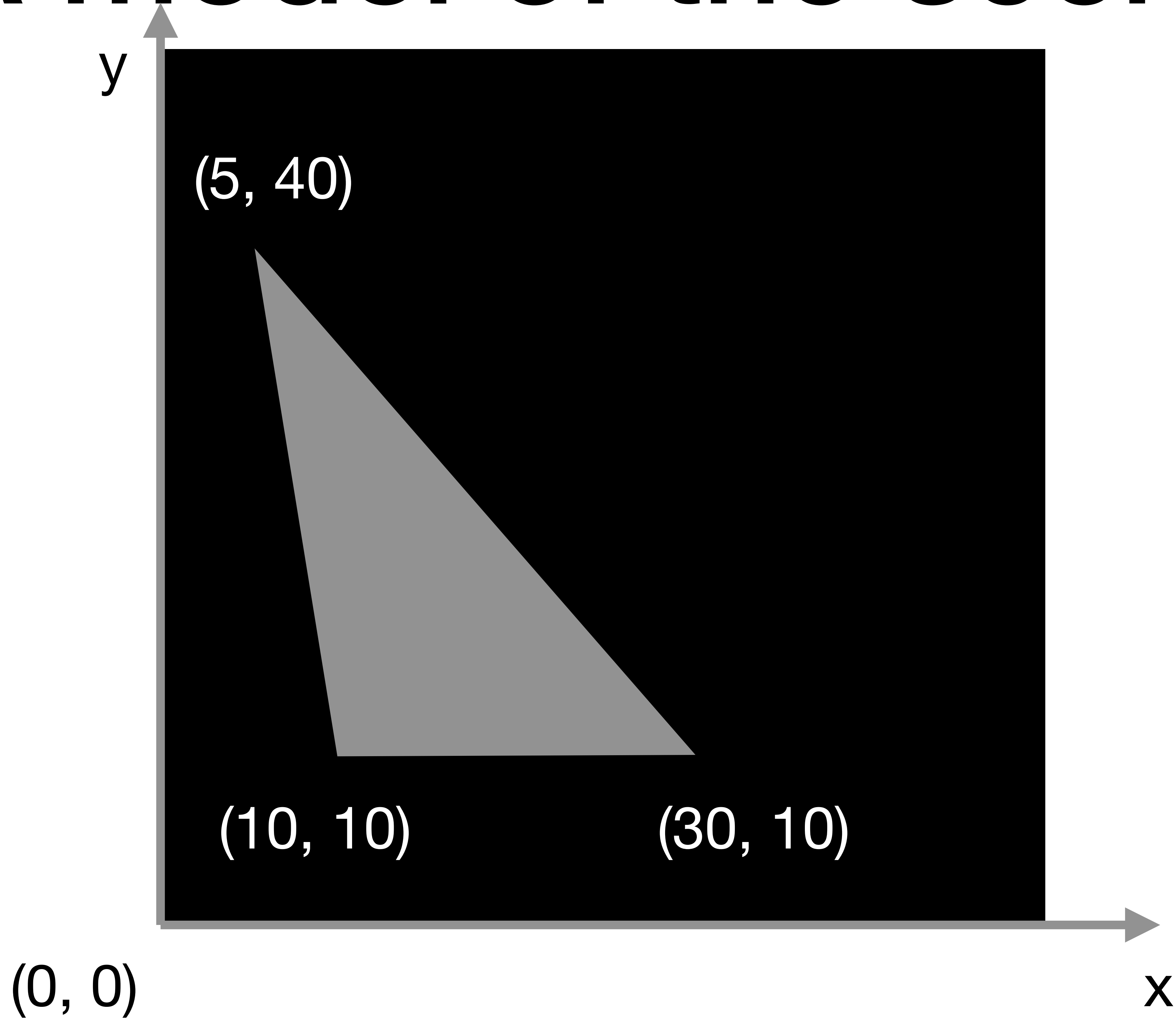


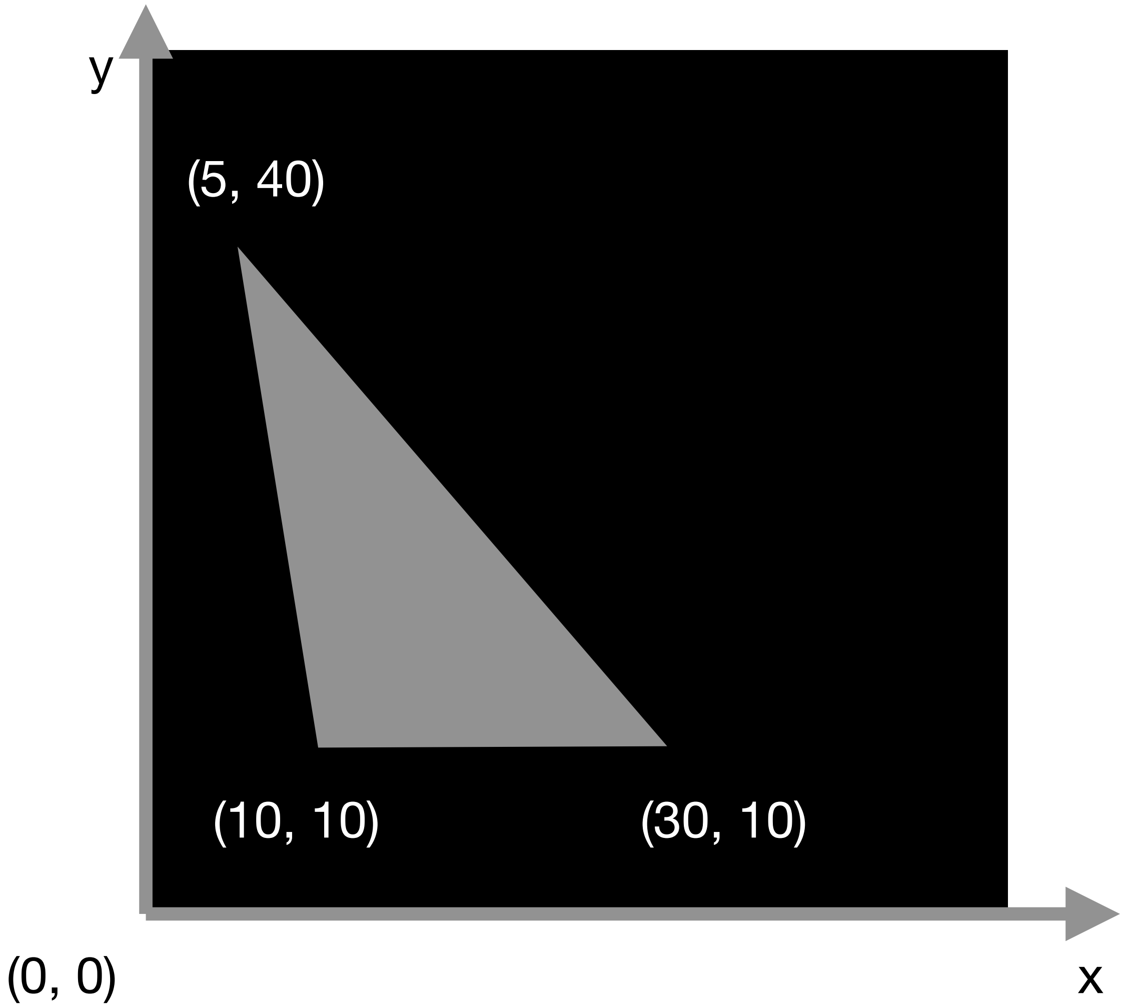
Image coordinates

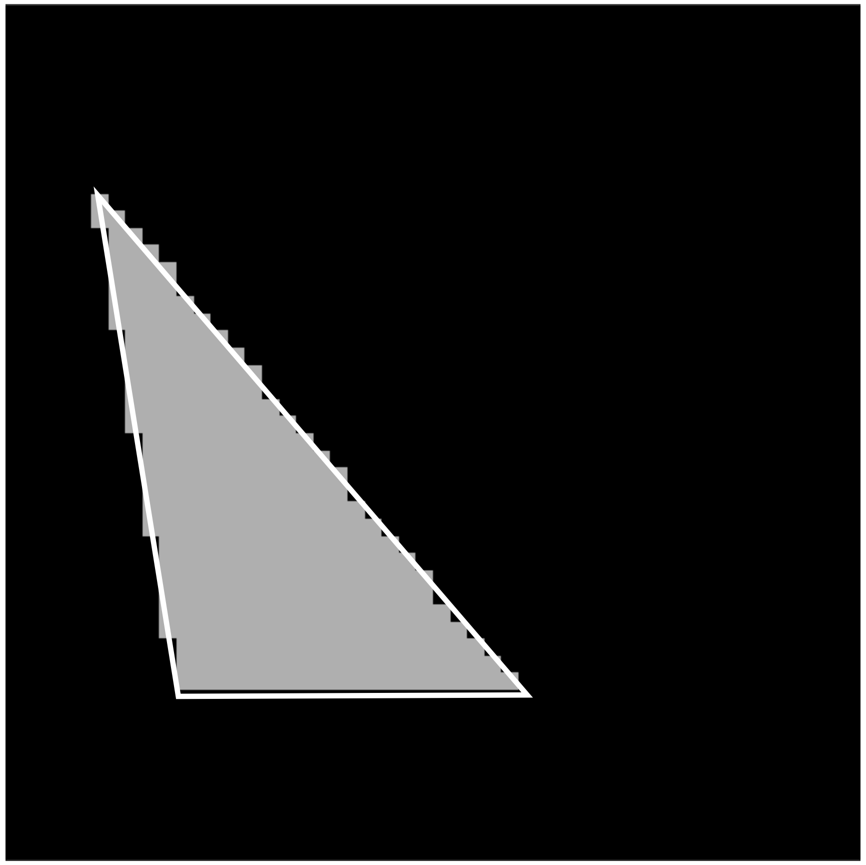
Problem: transform x, y position to i, j coordinates

A model of the scene



A0: Rendering (*Rasterizing*) a Triangle





Pseudocode:

```
for each pixel p:
  if p is inside triangle:
     p = color
```

Images in Julia: Demo

- Draw a rectangle on a canvas
- Demo colors
- See image_demo.jl

Vectors (whiteboard)

(See written notes)