Outline: 9/14

- Line algorithms (+ handout)
- Intro to HTML and JavaScript

Admin: Office Hours

- Lab 1
  - Monday 4-5pm (location TBD)
  - Tuesday 4-5pm (346 Ford)

TA hours Sun-Tues:
  - 7:30-9:30pm (241 Ford)
One more graphics application

Graphic Design Junction

AaBbCcDdEeFfGgHhIi
JjKkLmMmNnOoPpQqRr
SsTuUuWwXxYyZz
1234567890

graphicdesignjunction.com

www.crafthubs.com

smashinghub.com
Intro to HTML
(mostly used as a wrapper for JavaScript)
<!DOCTYPE html>
<html>
<head>
<title>Canvas Graphics</title>
</head>
<body onload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
Intro to HTML

(Hyper Text Markup Language)

```html
<!DOCTYPE html>
<html>
<head>
<title>Canvas Graphics</title>
</head>
<body onload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
Intro to HTML

(Hyper Text Markup Language)

```html
<!DOCTYPE html>
<html>
<head>
<title>Canvas Graphics</title>

HTML tags (start tag and end tag)

after webpage has loaded (text, images, etc), call this function

</head>
<body onload="init()">
  <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
Intro to HTML

(Hyper Text Markup Language)

```html
<!DOCTYPE html>
<html>
<head>
<title>Canvas Graphics</title>
</head>
<body onload="init()">
  <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```

HTML tags (start tag and end tag)

after webpage has loaded (text, images, etc), call this function

size of the window (you choose)
Intro to HTML

(Hyper Text Markup Language)

```html
<!DOCTYPE html>
<html>
<head>
<title>Canvas Graphics</title>
</head>
<body onload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```

HTML tags (start tag and end tag)

after webpage has loaded (text, images, etc), call this function

size of the window (you choose)

id: name the element, which we can refer to later
Intro to HTML

(Hyper Text Markup Language)

```html
<!DOCTYPE html>
<html>
<head>
<title>Canvas Graphics</title>
</head>
<body onload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```

- HTML tags (start tag and end tag)
- Indentation: not required, but good style
- size of the window (you choose)
- after webpage has loaded (text, images, etc), call this function
- id: name the element, which we can refer to later
HTML examples

Lists

- header 1 tag (h1-h6) exist, (large to small)
- unordered list (ol for ordered list)
- list element

<h1>My courses</h1>
<ul>
  <li>Graphics</li>
  <li>Calculus</li>
  <li>Yoga</li>
</ul>
**HTML examples**

**Lists**

- header 1 tag (h1-h6) exist, (large to small)

- unordered list (ol for ordered list)

- list element

**Tables**

- table row

- table data
**HTML examples**

**Lists**
- header 1 tag (h1-h6) exist, (large to small)
- unordered list (ol for ordered list)
- list element

**Tables**
- table row
- table data

This type of code usually goes after the head.
Intro to JavaScript
Intro to JavaScript

```html
<!DOCTYPE html>
<html>
<head>
<title>Canvas Graphics</title>

    <!-- JavaScript goes here (usually within the head) -->

</head>
<body onload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
```html
<!DOCTYPE html>
<html>
<head>
  <title>Canvas Graphics</title>
  <script>
    var canvas; // DOM object corresponding to the canvas
    var graphics; // 2D graphics context for drawing on the canvas

    function draw() {
      // draw on the canvas, using the graphics context
      graphics.fillText("Hello World", 10, 20);
    }

    function init() {
      canvas = document.getElementById("theCanvas");
      graphics = canvas.getContext("2d");
      draw(); // draw something on the canvas
    }
  </script>
</head>
<body onload="init()">
  <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
<script>
    var canvas;  // DOM object corresponding to the canvas
    var graphics;  // 2D graphics context for drawing on the canvas

    function draw() {
        // draw on the canvas, using the graphics context
        graphics.fillText("Hello World", 10, 20);
    }

    function init() {
        canvas = document.getElementById("theCanvas");
        graphics = canvas.getContext("2d");
        draw();  // draw something on the canvas
    }
</script>
</head>
<body onload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
Intro to JavaScript

Python:
```python
def init():
    pass
```

JavaScript:
```javascript
function init() {
    // DOM object corresponding to the canvas
    var canvas;
    // 2D graphics context for drawing on the canvas
    var graphics;

    function draw() {
        // draw on the canvas, using the graphics context
        graphics.fillText("Hello World", 10, 20);
    }

    function init() {
        canvas = document.getElementById("theCanvas");
        graphics = canvas.getContext("2d");
        draw(); // draw something on the canvas
    }
}
```

```html
<body onunload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
```html
<script>
var canvas; // DOM object corresponding to the canvas
var graphics; // 2D graphics context for drawing on the canvas

function draw() {
    // draw on the canvas, using the graphics context
    graphics.fillText("Hello World", 10, 20);
}

function init() {
    canvas = document.getElementById("theCanvas");
    graphics = canvas.getContext("2d");
    draw(); // draw something on the canvas
}
</script>
</head>
<body onload="init()">
    <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
Declare the variable canvas
Initialize canvas
Refer to canvas attributes
Example: canvas.width

```html
<script>
  var canvas; // DOM object corresponding to the canvas
  var graphics; // 2D graphics context for drawing on the canvas

  function draw() {
    // draw on the canvas, using the graphics context
    graphics.fillText("Hello World", 10, 20);
  }

  function init() {
    canvas = document.getElementById("theCanvas");
    graphics = canvas.getContext("2d");
    draw(); // draw something on the canvas
  }
</script>
</head>
<body onload="init()">
  <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
Intro to JavaScript

Python:  
```python
i = 1
```

JavaScript:  
```javascript
var i = 1;
```

```html
<html>
<head>
</head>
<body onload="init()">
  <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
Note: neither “var” nor “;” are strictly necessary, but good practice!

Python:

```python
i = 1
```

JavaScript:

```javascript
var i = 1;
```

```
<script>
  var canvas;  // DOM object corresponding to the canvas
  var graphics;  // 2D graphics context for drawing on the canvas

  function draw() {
    // draw on the canvas, using the graphics context
    graphics.fillText("Hello World", 10, 20);
  }

  function init() {
    canvas = document.getElementById("theCanvas");
    graphics = canvas.getContext("2d");
    draw();  // draw something on the canvas
  }
</script>
</head>
<body onload="init()">
  <canvas id="theCanvas" width="640" height="480"></canvas>
</body>
</html>
```
Intro to JavaScript

```html
<script>
  var canvas;  // DOM object corresponding to the canvas
  var graphics; // 2D graphics context for drawing on the canvas

  function draw() {
    // draw on the canvas, using the graphics context
    graphics.fillText("Hello World", 10, 20);
  }

  function init() {
    canvas = document.getElementById("theCanvas");
    graphics = canvas.getContext("2d");
    draw(); // draw something on the canvas
  }
</script>
```
Graphics functions

1) graphics.fillStyle = “red”;

2) graphics.fillRect(x, y, w, h);
Graphics functions

1) graphics.fillStyle = “red”;

2) graphics.fillRect(x, y, w, h);
Graphics functions

1) graphics.fillStyle = “red”;

2) graphics.fillRect(x, y, w, h);
1) `graphics.fillStyle = "red";`

2) `graphics.fillRect(x, y, w, h);`

For now: `graphics.fillRect(x, y, 1, 1);`
For Loops

Python:
for i in range(20, 40, 2):
    print(i)

JavaScript:
for (var i = 20; i < 40; i += 2) {
    window.alert(i);
}
Conditionals

Python:

```python
if x > 0:
    print("positive")
elif x < 0:
    print("negative")
else:
    print("zero")
```

JavaScript:

```javascript
if (x > 0) {
    window.alert("positive");
} else if (x < 0) {
    window.alert("negative");
} else {
    window.alert("zero");
}
```
Other Syntax

- Mod: %
- And: &&
- Or: ||
- Comment: //
- Always double check:
  - curly braces {}
  - indent based on {}
  - parenthesis ()
Debugging Demo
Poll Everywhere + Lab 1
What is the slope of a line with endpoints (50,30) and (70,30)?

- 0: 93%
- 1: 7%
- Infinity: 7%
What is the slope of a line with endpoints (50,30) and (50,70)?

When poll is active, respond at PollEv.com/saramathieso692
Text SARAMATHIESO692 to 22333 once to join

infinity 85%

0 12%
1 2%