



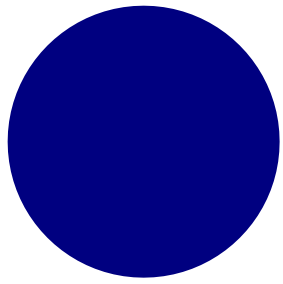
# **GEOG 178/258**

## **Week 9:**

**Serialization and Planar Graphs**

*mike johnson*





# Telling Java an Object should be serializable

Week

9

Part 1

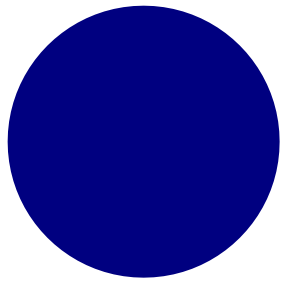
```
import java.io.Serializable;

public class Point implements Serializable {
    private double x, y;
    private String name;
    private boolean visited = false;
```

```
import java.awt.Color;

public class Polyline implements Serializable {
    // Define a arrayList which stores the point class objects
    private ArrayList<Point> points;

    // Define a boolean which stores whether or not the polyline is closed
    private boolean closed = false;
```



# Initializing a Menu

Week

9

Part 1

```
public GUI() throws IOException, URISyntaxException {
    // Feel free to ignore, this just calls the constructor of JPanel to enable DoubleBuffering to avoid flickering.
    super(true);

    // You always need a frame to place other components such as panels or buttons
    frame = new JFrame("TinyGIS");
    frame.setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE); // Taken care of by event handler instead
    frame.add(this);

    // Register the gui to listen to mouse events
    this.addMouseListener(this);
    this.addMouseMotionListener(this);

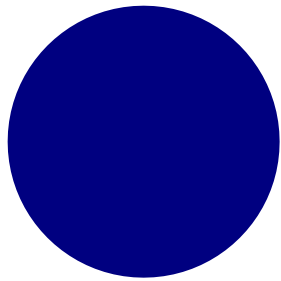
    // Set the (preferred) size of the panel
    setPreferredSize(new Dimension(699,446));

    // Load background image
    if (basemap == null) basemap = ImageIO.read(new File(TinyGIS.class.getResource("basemap.png").toURI()));

    // Create menu bar
    JMenuBar menuBar = new JMenuBar();

    // Create file menu
    JMenu fileMenu = new JMenu("File");
    openMenuItem = new JMenuItem("Open...");
    saveMenuItem = new JMenuItem("Save");
    saveAsMenuItem = new JMenuItem("Save As...");
    saveMenuItem.setEnabled(false);
    menuBar.add(fileMenu);
    fileMenu.add(openMenuItem);
    fileMenu.add(new JSeparator());
    fileMenu.add(saveMenuItem);
    fileMenu.add(saveAsMenuItem);
    openMenuItem.addActionListener(this);
    saveMenuItem.addActionListener(this);
    saveAsMenuItem.addActionListener(this);
}
```

1. Create a Menu bar
2. Add Options...
3. Enabled false
4. Add fileMenu to bar
5. Add options to fileMenu
6. Set actionListeners



# Defining File Open

```
// a button was pressed
public void actionPerformed(ActionEvent e) {
    if (e.getSource() == openMenuItem) {
        System.out.println("User initiated open function.");

        JFileChooser fileChooser = new JFileChooser();
        fileChooser.setDialogTitle("Open");
        int fileChooserResult = fileChooser.showOpenDialog(frame);

        if (fileChooserResult == JFileChooser.APPROVE_OPTION) {
            File openPath = fileChooser.getSelectedFile();

            try {
                FileInputStream fileIn = new FileInputStream(openPath.getAbsolutePath());
                ObjectInputStream objIn = new ObjectInputStream(fileIn);

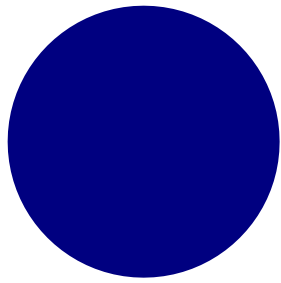
                saveFile = (SaveFile) objIn.readObject();
                pp = saveFile.getPp();
                pl = saveFile.getPl();

                objIn.close();
                fileIn.close();

                saveFilePath = openPath.getAbsolutePath();
                saveMenuItem.setEnabled(true);
                fileSaved();

                repaint();

                System.out.println("Opened file: " + saveFilePath);
            } catch (FileNotFoundException i) {
                System.out.println("Open failed!");
                JOptionPane.showMessageDialog(frame, "Could not find a file!", "No File Found", JOptionPane.ERROR_MESSAGE);
            } catch (IOException i) {
                System.out.println("Open failed!");
                JOptionPane.showMessageDialog(frame, "The file you selected is not compatible with TinyGIS.", "Invalid File", JOptionPane.ERROR_MESSAGE);
            } catch (ClassNotFoundException i) {
                System.out.println("Open failed!");
                JOptionPane.showMessageDialog(frame, "The file you selected is not compatible with TinyGIS.", "Invalid File", JOptionPane.ERROR_MESSAGE);
            }
        } else {
            System.out.println("Open aborted!");
        }
    }
}
```



# Saving a File

```
} else if (e.getSource() == saveAsMenuItem) {
    System.out.println("User initiated save as function.");

    JFileChooser fileChooser = new JFileChooser();
    fileChooser.setDialogTitle("Save As");
    File file = new File(System.getProperty("user.home")+java.io.File.separator+"MyData.tgis");
    fileChooser.setSelectedFile(file);
    int fileChooserResult = fileChooser.showSaveDialog(frame);

    if (fileChooserResult == JFileChooser.APPROVE_OPTION) {
        File savePath = fileChooser.getSelectedFile();

        try {
            FileOutputStream fileOut    = new FileOutputStream(savePath.getAbsolutePath());
            ObjectOutputStream objOut    = new ObjectOutputStream(fileOut);

            saveFile = new SaveFile(pp, pl);
            objOut.writeObject(saveFile);

            objOut.close();
            fileOut.close();

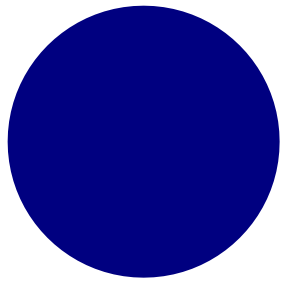
            saveFilePath = savePath.getAbsolutePath();
            saveMenuItem.setEnabled(true);
            fileSaved();

            System.out.println("Saved as file: " + saveFilePath);
        } catch(IOException i) {
            System.out.println("Save as failed!");
            JOptionPane.showMessageDialog(frame, "There was an error in saving your data.", "Save Error", JOptionPane.ERROR_MESSAGE);
        }
    } else {
        System.out.println("Save as aborted!");
    }
}
```



# Sub-code (added here to save space)

```
3 import java.io.Serializable;
4
5 public class SaveFile implements Serializable {
6     private Polypoint pp;
7     private Polyline pl;
8
9     public SaveFile (Polypoint pp, Polyline pl) {
10         this.pp = pp;
11         this.pl = pl;
12     }
13
14     public Polypoint getPp() {
15         return pp;
16     }
17
18     public void setPp(Polypoint pp) {
19         this.pp = pp;
20     }
21
22     public Polyline getPl() {
23         return pl;
24     }
25
26     public void setPl(Polyline pl) {
27         this.pl = pl;
28     }
29 }
```

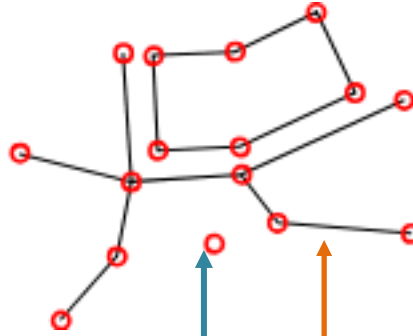


# Homework

Week

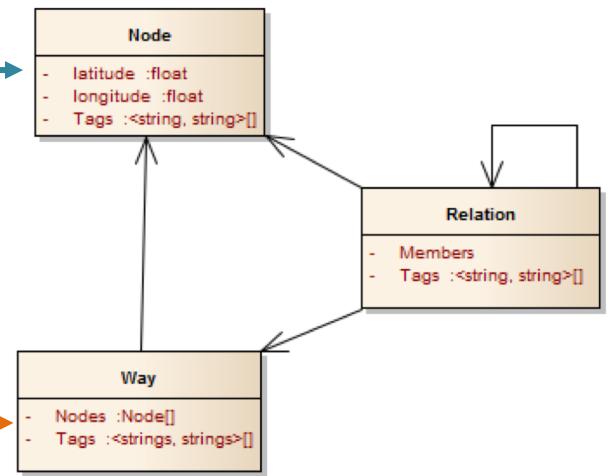
9

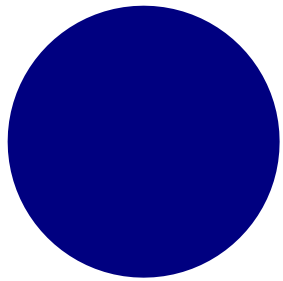
Part 2



POI!

PATH!





# OSM (Nodes)

Week

9

Part 2

The screenshot shows the OpenStreetMap interface. A red box highlights a node popup for 'The Home Depot (1672439282)'. The popup contains the following information:

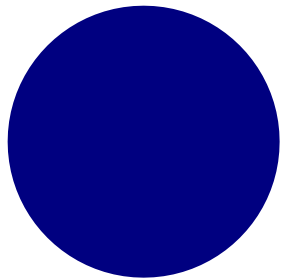
- Fix retail.
- Edited 3 months ago by [escallic](#)
- Version #2 - Changeset #65078927
- Location: 34.4276937, -119.8706973

Below the popup is a table of tags:

<a href="#">addr:city</a>	Goleta
<a href="#">addr:housenumber</a>	6975
<a href="#">addr:postcode</a>	93117
<a href="#">addr:street</a>	Marketplace Drive
<a href="#">name</a>	The Home Depot
<a href="#">phone</a>	+1-805-961-4746
<a href="#">shop</a>	doityourself

At the bottom of the popup, there are links for [Download XML](#) and [View History](#).





Week

9

Part 2

# OSM (Ways)

**Way: Storke Road (231846783)** ✕

retail

Edited 4 months ago by [escallic](#)  
Version #7 · Changeset #64458692

Tags

cycleway	lane
highway	secondary
lanes	5
name	Storke Road
oneway	yes
surface	asphalt
tiger:cfcc	A45
tiger:county	Santa Barbara, CA
tiger:name_base	Storke
tiger:name_type	Rd
turn:lanes	left left[none] none

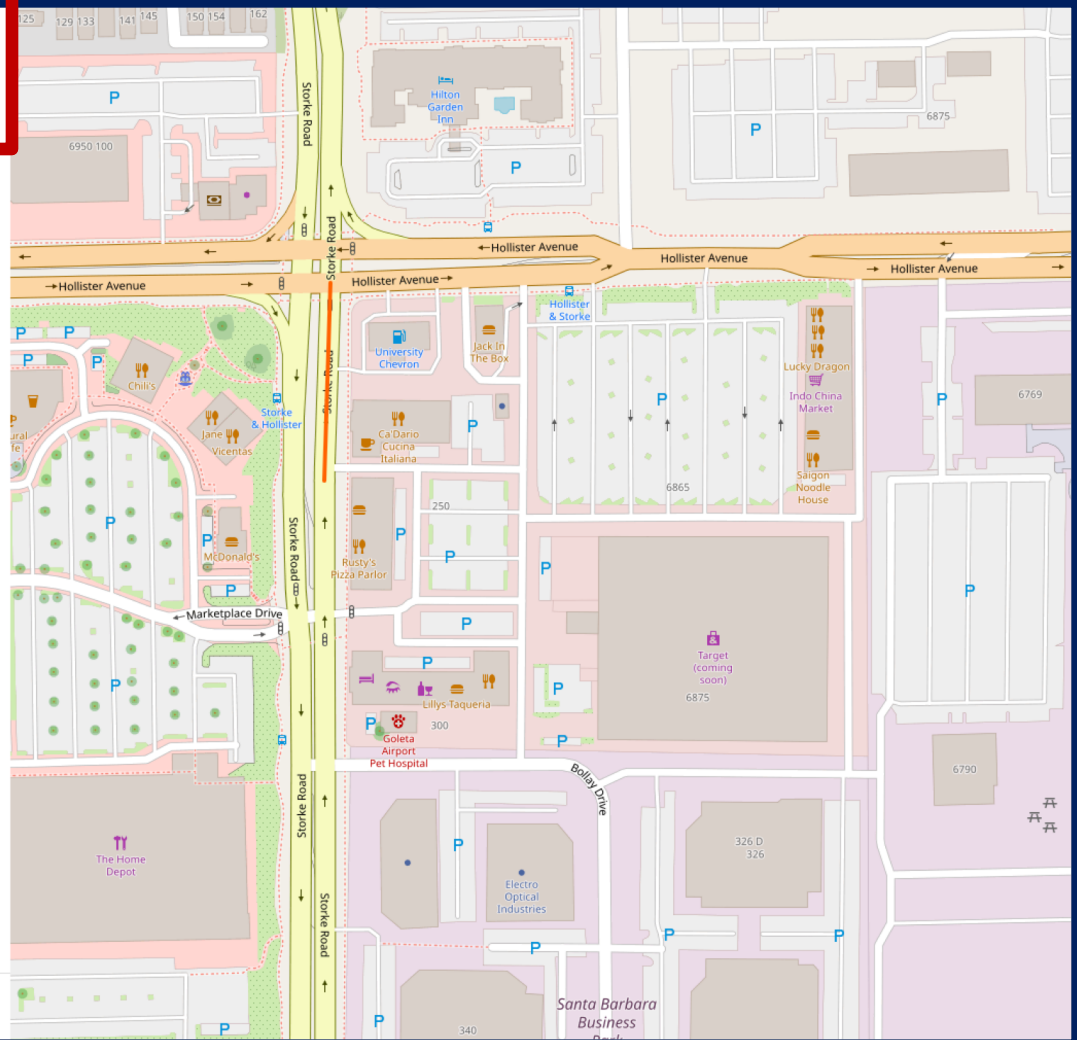
Part of

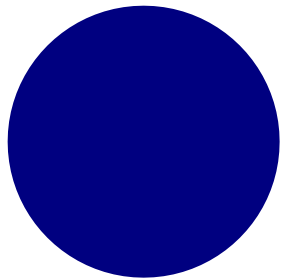
- Relation Coast Bicycle Route (6637813) (as forward)
- Relation Winchester Canyon (1105800) (as forward)
- Relation UCSB (1101228) (as forward)
- Relation Goleta (1101224) (as forward)
- Relation UCSB Express (1105825) (as forward)
- Relation SBCC/UCSB Express (1101565) (as forward)

Nodes

- 5416803482 (part of way — Storke Road (561820425))
- 3771334361 (part of way — 373714167)
- 6058746615 (part of way — 643955894)
- 5416803481
- 5416803475 (part of way ..... 542875436)
- 165341872 (part of ways — Storke Road (561820410), — Hollister Avenue (69728457), and — Hollister Avenue (70346777))

[Download XML](#) · [View History](#)





Week

9

Part 2

# OSM Routing

University of California, Santa Barbara, Isla Vista

Car (OSRM) Go

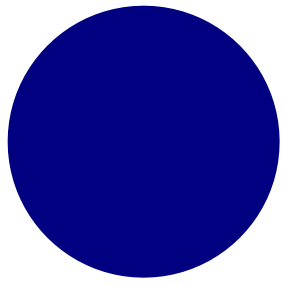
[Reverse Directions](#)

**Directions** ×

Distance: 4.1km. Time: 0:07.

1. Start on unnamed road 40m
2. Turn left onto unnamed road 50m
3. At the end of the road turn right onto **Marketplace Drive** 50m
4. Turn left onto **Storke Road** 170m
5. Turn right onto **Hollister Avenue** 900m
6. Turn right onto **South Los Carneros Road** 900m
7. Turn left onto **Mesa Road** 1300m
8. Turn right onto **Ocean Road** 400m
9. Continue on unnamed road 50m
10. At roundabout take 1st exit onto unnamed road 90m
11. Exit roundabout onto unnamed road 200m
12. Reach destination

Directions courtesy of [FOSSGIS Routing Service](#)



# Our Job: Build a Graph of ways and nodes

Week

9

## Part 2

To go from D to F

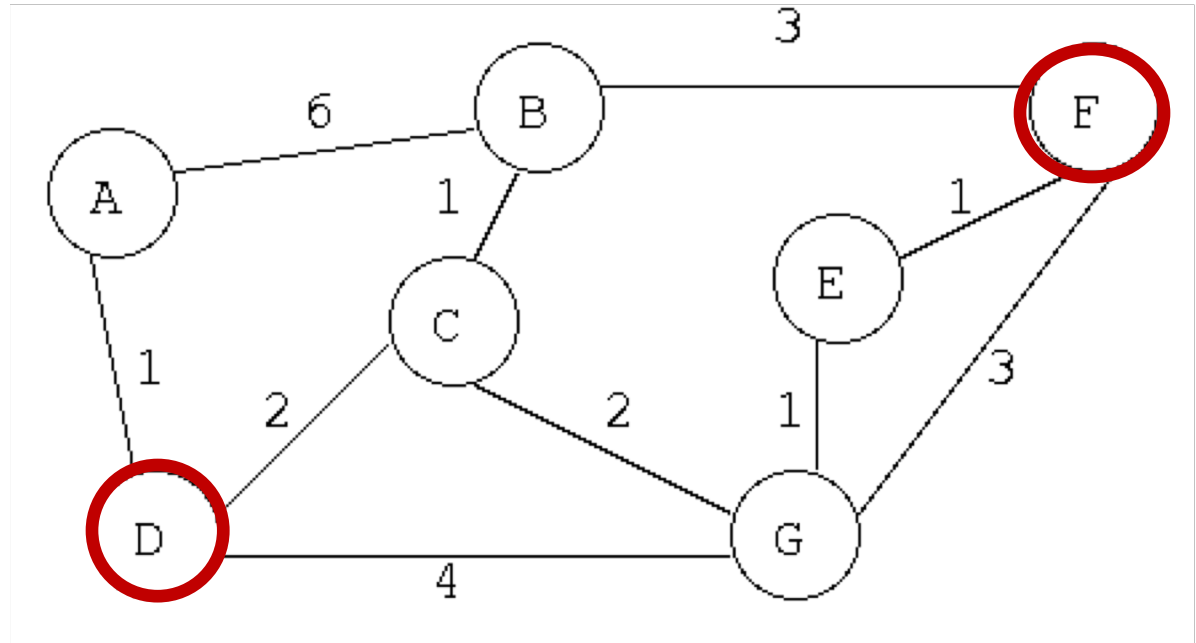
DABF = 10

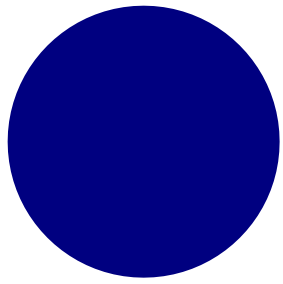
DCBF = 6

DCGEF = 6

DGF = 7

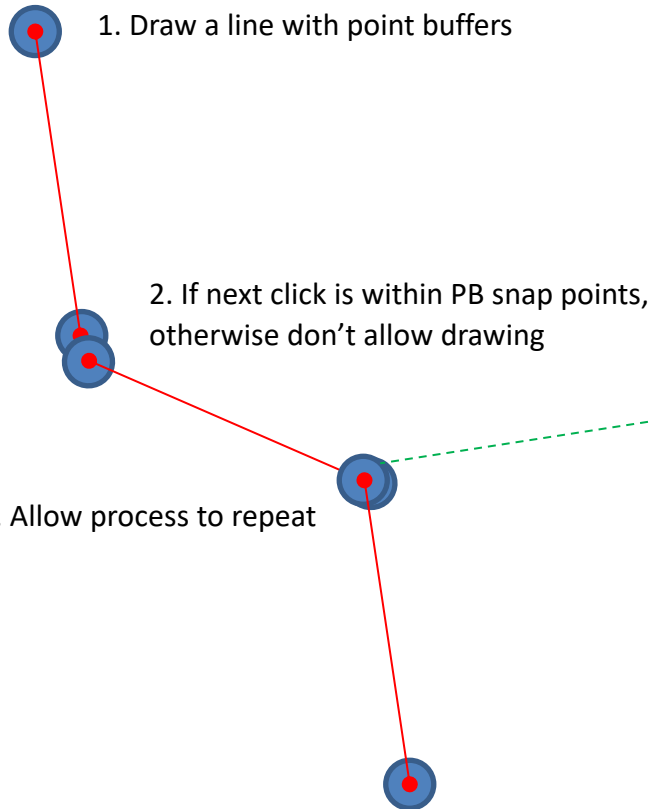
DGEF = 6





# Minimum Example:

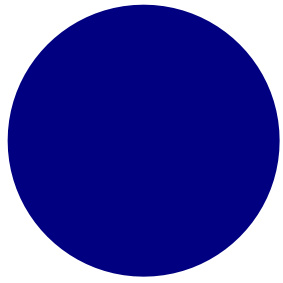
## 1A. Define line class (serialize?)



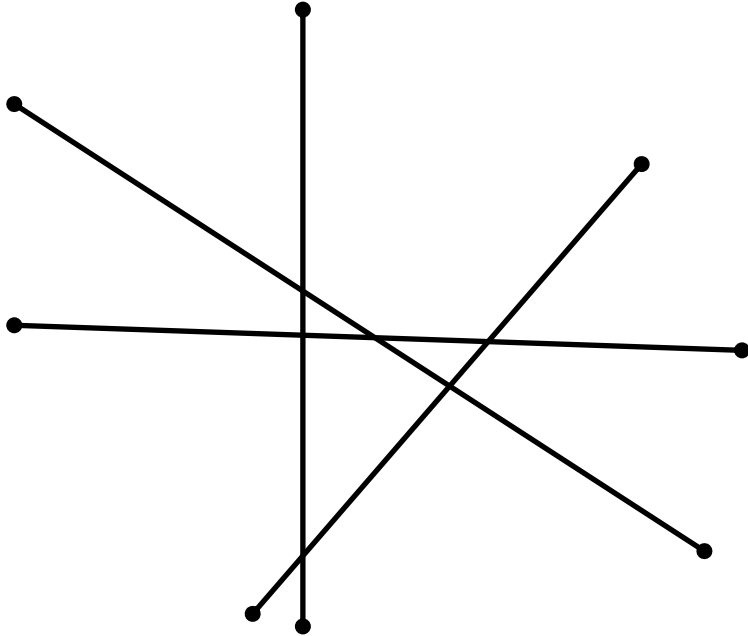
## 1B. Add POI \*258

find nearest node and generate line

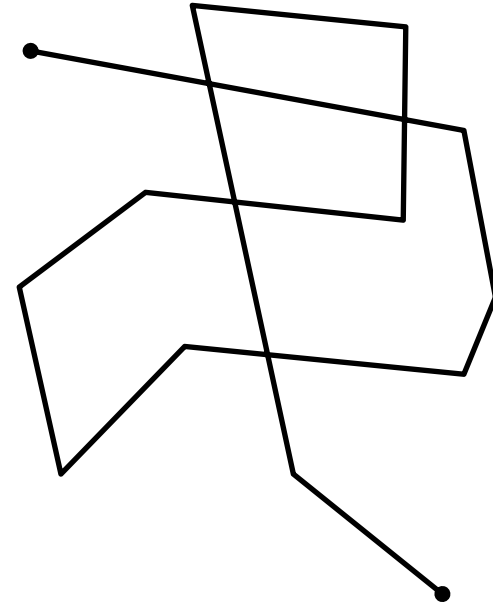
● Home Depot



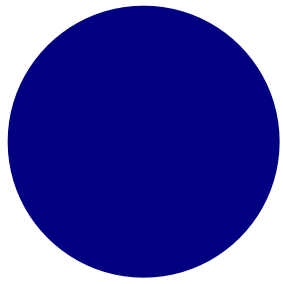
## More Difficult: Graph from lines



*Currently: 4 line segments and 8 Points  
Should be: 16 ways and 26 nodes*



*Currently: One PolyLine (13 Points)  
Should be: 17 nodes, 20 ways*



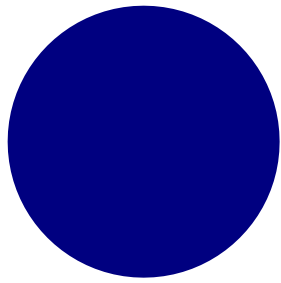
# Finding Intersecting Nodes

Week

9

Part 2

Forms for the Equation of a Line		
Slope-Intercept	$y = mx + b$	$m$ is the slope $b$ is the y-intercept
Point-Slope	$y - y_1 = m(x - x_1)$	$m$ is the slope $(x_1, y_1)$ is a point on the line
Standard Form	$ax + by = c$	$a$ is positive
Intercept Form	$\frac{x}{a} + \frac{y}{b} = 1$	$a$ is the x-intercept $b$ is the y-intercept
Vertical	$x = a$	Vertical line with $a$ as the x-intercept
Horizontal	$y = b$	Horizontal line with $b$ as the y-intercept



# Finding the Intersection (Theory)

Express 2 Lines:

$$\begin{aligned} a_1X + b_1Y &= C_1 \\ a_2X + b_2Y &= C_2 \end{aligned}$$

$$\begin{bmatrix} a_1X & b_1Y \\ a_2X & b_2Y \end{bmatrix} = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}$$

Calculate Determinant:

$$|A| = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} = (a_1*b_2 - a_2*b_1) \text{ //determinant}$$

$$A^{-1} = \frac{1}{|A|} \begin{bmatrix} b_2 & -b_1 \\ -a_2 & a_1 \end{bmatrix}$$

Re-order Equations:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{|A|} \begin{bmatrix} b_2 & -b_1 \\ -a_2 & a_1 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix}$$

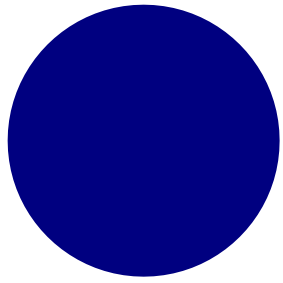
$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{|A|} \begin{bmatrix} b_2C_1 & -b_1C_2 \\ -a_2C_1 & a_1C_2 \end{bmatrix}$$

Solve for X, Y:

$$X = (b_2C_1 - b_1C_2) / (a_1*b_2 - a_2*b_1)$$

$$Y = (a_1C_2 - a_2C_1) / (a_1*b_2 - a_2*b_1)$$

new Point (x, y)



# Finding the Intersection (in Java)

Solve for X, Y:

$$X = (b2 * C1 - b1 * C2) / (a1 * b2 - a2 * b1)$$
$$Y = (a1 * C2 - a2 * C1) / (a1 * b2 - a2 * b1)$$

new Point (x, y)

Where  $a1 = \Delta X_{Line1}$

Where  $b1 = \Delta Y_{Line1}$

Where  $C1 = a1 * X_{Line1, Point1} + b1 * Y_{Line1, Point1}$

Where  $a2 = \Delta X_{Line2}$

Where  $b2 = \Delta Y_{Line2}$

Where  $C2 = a2 * X_{Line2, Point1} + b2 * Y_{Line2, Point1}$



# In Class Example

```
Duplicates();  
);
```

```
ldGraph() {
```

```
    = this.buildN  
    ew Lines();
```

```
    i < this.size
```

```
    es = new Node  
    des = new Pol  
    ouble> dist =
```

```
    0; k < nodes
```

```
    .get(k).getLi  
    nodes.add(node
```

```
int(this.get(
```

```
    0; j < tmpNo
```

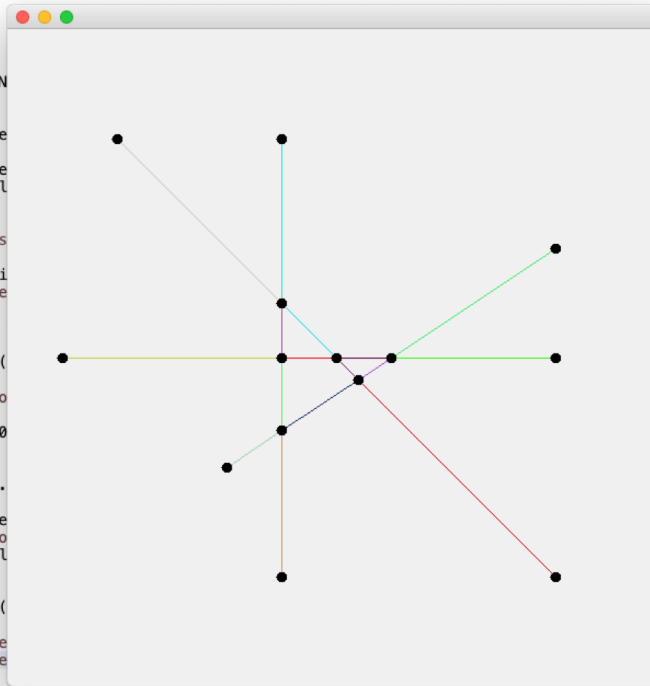
```
(oNodes.get(0
```

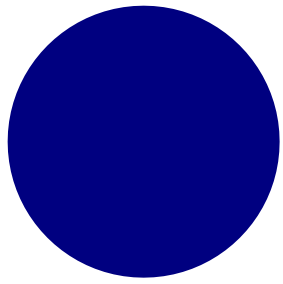
```
    0; t < dist.
```

```
    x = dist.inde  
    ddPoint(tmpNo  
    (index, Doubl
```

```
int(this.get(
```

```
    0; b < oNode  
    ew Line(oNode
```





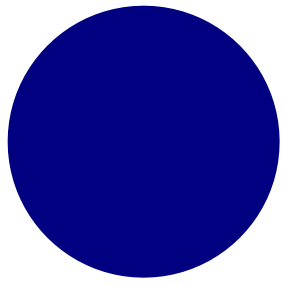
# HW Hints

Week

9

## Homework Hints

1. Can you turn your path (polyline) into type 'Lines'
  - Should path be a class?
2. Can you create a class of 'Paths' to store multiple paths
3. Can you create buttons to execute 'build nodes' and 'build graph'



# Rubric

Week

9

## Rubric

GEOG 178	Points
Can I import your code w/o modification?	2
Can I create a “planar” network (classes)	2
Can I create a “planar” network (interface)	2
Can I save a network?	2
Can I read the network back in?	2

GEOG 278	Points
Can I import your code w/o modification?	2
Can I create a “planar” network (classes)	1
Can I create a “planar” network (interface)	2
Can I save a network?	1
Can I read the network back in?	2
Can I connect a POI to the network?	2