Problem 1

In this exercise you'll read in two `Points`, make a `Line`, compute some attributes, and then output those values. Given a `Point` struct containing two floats,

```
struct Point { float x; float y; };
```

create a `Line` struct using two `Point` structs. The `Line` struct should contain the following elements:

1. Two `Point` coordinates
2. `float length`
3. `float slope`
4. `float y_intercept`

You'll need to write three functions to compute:

1. `float compute_slope(Line l)`
2. `float compute_y_intercept(Line l)`
3. `float compute_length(Line l)`

and assign these to the `Line` for each `Point` set (two coordinates). Additionally, write one more function, `printLineStats(Line l)`, to print out the `Line`'s information in this format (using, *e.g.,* the first set of coordinates from the file):

```
Coordinate Set 1
First Point  = (5.0, 3.0)
Second Point = (-1.0, 6.0)
Slope        = -0.5
y-intercept  = 5.5
Line length  = 6.7082
```

Your program should read inputs from the file `coordinates.txt` located on the ISP server in the directory:

```
/home/kschmidt/LectureExercises/HW05input
```

Copy this file to your own home directory using the command (note the period at the end):

```
cp /home/kschmidt/LectureExercises/HW05input/coordinates.txt .
```

and write your output to a file called `LinearAnalysis.txt`. You won't need to submit the output file with your homework, only the source for your code. There will be several lines in the `coordinates.txt` file, each containing two coordinates in the format:

```
x1-value y1-value x2-value y2-value
```

Note: We will NOT be testing for lines parallel to the x- or y-axis.