Instructions.   This exam has two questions. You have 50 minutes.

Resources.   This exam is open notes, open book, and open to the web—but you may only access the course website, booksite, and Piazza. You may not use Google or another search engine. You may not communicate with anyone during this exam in any way (e.g., talking, messaging, email, etc.).

Submit.   When you are done, submit your program using the submit link on the Meetings page.

Grading.   Your program will be graded on correctness, design, efficiency, and clarity including comments. You will receive partial credit for a program that correctly implements some of the required functionality. You will lose a large number of points if your program does not compile.

Discussing this exam.   Due to travel for extracurriculars and sports, some of your peers will take this exam next week. Do not discuss exam contents with anyone who has not taken the exam.

This paper.   Fill in the information below, then transcribe and sign the Honor Code pledge.

NAME: _____

NETID: _____

PRECEPT: _____

EXAM ROOM: _____

"I pledge my honor that I will not violate the Honor Code during this examination."

_____

_____

SIGNATURE: _____

Overview.   It seems like we've had more rain than usual this year. Let's investigate. Write a program, `Rainfall.java`, that reads in integers from `StdIn` and outputs their average, formatted with 2 digits after the decimal. Stop reading either when the value 999 is input or when there is no more input.

Example.   If the input is 1 2 3 999 4 5, the output should be the average of the first three values, 2.00.

Input and output.   The input consists of integers on `StdIn`. Print a double with 2 digits after decimal.

Data cleaning.   Treat all negative values as zeros. For example, the input 1 -3 999 would yield 0.50.

Edge cases.   Have you considered the following?

- What happens if the input is just 999? Your output should be 0.00.

- What happens if the input contains no 999s? Average all the values, treating negatives as zeros.

- What happens if the input contains two or more 999s? Do not read any input after the first 999.

Overview.   How does precipitation change over time? Write a program, `Precipitation.java`, that plots the annual precipitation since 1900 by reading in data from `StdIn` and creating a line plot.

Input.   Read in pairs of years (as integers) and annual precipitation (as doubles) from `StdIn`.

Output.   Plot a line between each consecutive pair of values where the year is the x-coordinate and the precipitation is the y-coordinate. Set x-scale to 1900-2017. Set y-scale to 15-75.
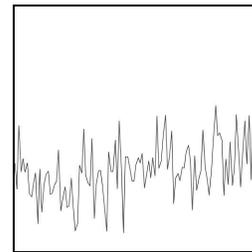
```
> more princeton.txt
1900 43.31
1901 53.89
...
```

```
> more seattle.txt
1900 36.48
1901 30.18
...
```

```
> java-introcs Precipitation < princeton.txt
```
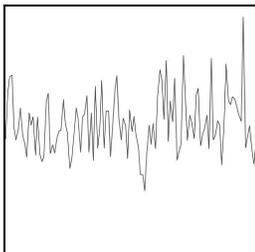
```
> java-introcs Precipitation < seattle.txt
```


Hint.   Call `StdDraw.line()` for every consecutive pair of points. For example, for `princeton.txt`, your first call to `StdDraw.line()` should use the first 4 values as arguments: 1900, 43.31, 1901, 53.89.