CIS 313, Intermediate Data Structures
Winter 2019

CIS 313 Lab 0

Due: Wednesday, January 16, 2019 at 4:55pm

This lab is an introduction into lab procedure and a review of Python

Overview

Write a program that reads a series of pairs of integers (X, Y) and prints pairs GCD(X,Y) and LCM(X,Y) where:

- GCD(X,Y) is the Greatest Common Divisor of X and Y
- LCM(X,Y) is the Least Common Multiple of X and Y

The purpose of this assignment is:

- Practice using standard input and output.
- Learn how to submit your work.

You should write your program in Python 2.7.15 (the default version on the ix-dev server) only.

You may not import any math libraries to compute LCM or GCD for you.

Input Description

Your program should accept an input file from the command line. The first line will contain an integer N, which is the number of (X, Y) pairs in the file. Each of the N lines contains two integers X and Y, separated by a space. For example *inSample.txt* below will be provided.

```
5
4 2
8 9
143 247
1 1
10 10
```

Output Description

For each of the pairs X Y output GCD(X, Y) and LCM(X, Y) on a line, separated by a single space. For example, using the sample input above, your program should output:

```
2 4
1 72
13 2717
1 1
10 10
```

Testing Protocol

We will test your program by running it from the command line on the department's development server. You will need to pass in the name of the file as an argument - do **not** encode the name of your input file form within your program. We will run your program with several different test files.

Examples of the commands we will use to test your program look like the following. Here *lab0.py* is a name of a turned-in program and *inSample.txt* is the sample test file. At the command line we might say something like

```
python lab0.py inSample.txt
```

Testing

Your code must be executable from the department's ix-dev server. If your code does not run in this environment you will lose a significant proportion of the available points for an assignment.

These steps tell you how to access your ix-dev account, and how to copy and run files from your localhost to the server.

1. Set up your ix-dev account with Cheri in the front office of the CIS building.
2. Once you have access, you need to log into ix-dev. From the command line run:
   `ssh yourUserName@ix-dev.cs.uoregon.edu`
3. Navigate to your workspace: (if you do not have these folders, use command mkdir to create them)
   `cd Documents/workspace`
4. Create a cs313 folder by running:
   `mkdir cs313`
5. Navigate into cs313 and create another folder lab0
   note: to find out the path to this folder navigate into the newly created lab0 folder and run:
   `pwd`
6. exit out of ssh by running the command "exit"

7. Now copy your .py files to your ix-dev account. While in the folder containing your .py files for this project on your personal machine run:

   ```
   scp * yourUserName@ix-dev.cs.uoregon.edu:~/Documents/workspace/cs313/lab0
   ```

8. ssh back into `yourUserName@ix-dev.cs.uoregon.edu`
9. Run your code for lab0 from the server as you would on your local computer.

Note: You will be provided at least one instance of an example input and output file. We will test your program with a mode diverse set of inputs. However, it is up to you to test and debug your program with your own set of input program. This way, you will gain experience in generating tests and identifying corner cases. If your code completes for the given input file, make sure that it will also accept and complete for other files just as well.

Submission

You are provided with skeleton code (lab0.py) that you may use, which should streamline the assignment. Submit **only** your .py file(s) via Canvas.

Name your program your first name, then last name, then _lab0.py. For example, I would name my program MattHall_Lab0.py. If you do not name your file this way you will lose points.

Grading

Typically, half of your grade will be determined by attempting to implement the correct data structure, and half will be on correctness. This assignment only has the correctness component. There will be 50 points possible for this assignment. Two points are assigned for each of the 25 test cases for this assignment. 5 of the test cases are given to you above, and 20 are hidden.