CS 260
Spring 2015
Lab 1

- **Please read and follow all instructions very carefully so that you do not lose points!**
- Write a Python3 program named "lab1.py" that operates as described below.
- Read from standard input stream (keyboard), and write to standard output stream (screen).
- The input will contain two integer values, either on the same line or on two consecutive lines.
- The output will consist of two lines, as follows:
    - The sum of the two input integers.
    - The product of the two input integers.

  See the example output files for precise output format.
- Example: If "input0.txt" is the following file:

  ```
  23 45
  ```

  Then this command should produce the output file "output0.txt" as shown below:

  ```
      python3 lab1.py < input0.txt > output0.txt
  ```

  ```
  sum = 68
  product = 1035
  ```
- Example: If "input1.txt" is the following file:

  ```
  54
  32
  ```

  Then this command should produce the output file "output1.txt" as shown below:

  ```
      python3 lab1.py < input1.txt > output1.txt
  ```

  ```
  sum = 86
  product = 1728
  ```

The following rules apply to all projects in this course:

- We now provide a virtual machine so you can test your program in the same environment where it will be graded. This server is "`cs260.ua.edu`". Regardless where you primarily develop your program, you must use this server to test your projects before submitting. The primary purpose of this lab is to help you understand the logistical issues related to using the server and submitting your future projects in this course. Login to the server using your Bama userid and password. You might first need to install the SSH secure shell and file transfer clients, for example at http://helpdesk.ua.edu/~recovery/ssh.htm. Also, if you wish to access the server from off-campus, you'll need to install a VPN client, which is available at https://mybama.ua.edu on the Tech tab.
- During grading, your program will be tested on the `cs260.ua.edu` server. Example scripts and input/output files will be provided on this server in the /projects directory.
- We provide a common platform that all students must use to test their programs before submitting, and we will use this same platform to evaluate each submitted program. Your program will NOT be evaluated based on how it performs in any other environment, because we will not attempt to replicate different environments used by each individual student.

- In previous semesters, some students did not follow the instructions and/or did not adequately test their programs before submitting, so they lost points for errors that they could have easily detected and corrected. This semester, it is required to test the final version of your program on the server before you submit it. All the scripts and input/output files are provided on the server in the /projects directory. When you run the script, it will produce a file “checksums.txt” that you must submit along with your program. When grading your project, we will use the contents of this file to verify that you actually did test the final version of your program on the server.
- You are permitted to verbally discuss ideas with classmates or other people, but sharing any code is strictly forbidden. **Do not look at your classmates’ or anybody else’s code, and do not show anybody your code!** However, you are encouraged to verbally discuss the projects with your classmates, provided nobody involved is looking at any code during the discussion.
- You must write all your own code for each project. Do not take code from any other source. If you happen to find a similar program online or in a book, you are NOT permitted to copy portions of that program into your program. The purpose of the projects is to learn how to write programs for data structures and algorithms, not to practice googling for solutions.
- I**f your code is too similar to another student's code or to anybody else’s code, you will receive an invitation to discuss the situation in the Dean's office.** We use the MOSS system which detects plagiarism based on programs that are too similar. The penalty for plagiarism is typically a score of 0 on the project or a grade of F in the course.
- You should strive to make each program as efficient as possible. During testing, we will allow each program to run for a reasonable amount of time, but if your program takes much longer to run than we think it should, we will eventually decide your program is too inefficient (or possibly in an infinite loop), and in this case we may terminate the execution of your program before it ends.
- Place all your program’s source files and the file “checksums.txt” (generated by the script) into a single zip file, give it a name of the form LastName_FirstName.zip (example: Doe_Jane.zip), and upload this zip file using Blackboard. Make sure your zip file includes the latest version of all your source files. Your zip file should contain the file “chesksums.txt” and all the source files needed to run your program, and these files should all be located at the top level of your zip file. The zip file should not contain any other subfolders. Do not submit projects via email.