Homework 04

For problems that require you to write a script, save your script in a directory named `scripts/` in the top of the homework directory. Write your script assuming the data file(s) it will use are in the current working directory.

1. The homework directory contains a subdirectory named `random/` which contains several files with random numbers in them. For the following problems, construct a single pipeline that will perform the required task and save this pipeline in a script that can be executed to perform the task. Your script should only contain a sheband and your pipeline. For example, if asked to write a pipeline to print the first 10 lines of column 1 in a comma-separated file named `data-01.csv`, you would create a file that contained the following

   ```
   #! /bin/bash
   cat data-01.csv | head | cut -d ',' -f 1
   ```

   You should also run the script and note its output. The Blackboard quiz will ask you to enter some of the information your scripts produce.

   1. Write a pipeline that will print the number of files in the current directory. Save your pipeline in a file named `Pipeline-1.sh`.

   2. Write a pipeline that will print the largest value in column 2 of `random-279.txt`. Save your pipeline in a file named `Pipeline-2.sh`.

   3. Write a pipeline that will print the minimum value in column 1 of `random-425.txt`. Save your pipeline in a file named `Pipeline-3.sh`.

   4. The files in this directory do not have the same number of lines. Write a pipeline that will print the name (just the name) of the file with the most lines. Save your pipeline in a file named `Pipeline-4.sh`.

   5. Write a pipeline that will print the maximum value in column 2 of *all files in the directory*. That is, your script will print a single number that is the maximum value found in column 2 of all files in the directory. Save your pipeline in a file named `Pipeline-5.sh`.

2. Write a shell script named `csv2txt` that takes two arguments. The first argument is the name of a comma-separated data file (a file that uses commas to separate data columns). The second argument is the name of a file your script will create that is a space-separated version of the csv file. So, for example, if a comma-separated file named `data-01.csv` exists, then `./csv2txt data-01.csv data-01.txt` should create a file named `data-01.txt` that contains the same data as `data-01.csv`, but separated by a single space instead of a comma.