SD212 12-Week Exam Practice (UPDATED)

1. Suppose you have two files `cities.csv` and `precip.csv` that look like this:

`cities.csv`

```
city,state,population,mayor
Phoenix,AZ,1680992,Kate Gallego
Dansville,NY,4433,Dennis Mahus
Ithaca,NY,32108,Laura Lewis
Frederick,MD,78171,Michael O'Connor
Annapolis,MD,40812,Gavin Buckley
```

`precip.csv`

```
city,yearly precipitation
Ithaca,37.3
Annapolis,47.27
Phoenix,8.03
Frederick,40.49
```

Write a Python program that reads in these two files to create a new file `cityrain.csv` that looks like below. (Ordering of the rows and number of decimal digits doesn't matter, but notice that Dansville isn't there!)

`cityrain.csv`

```
city,state,population,monthly precipitation
Phoenix,AZ,1680992,0.67
Ithaca,NY,32108,3.11
Frederick,MD,78171,3.37
Annapolis,MD,40812,3.94
```

2. Consider a file `badcities.csv`, similar to the last problem but with some issues:

badcities.csv

```
city,state,population,mayor
Dansville,NY,4433,Dennis Mahus
San Marino,,33745,Luca Beccari
Ithaca,NY,32108,Laura Lewis
New York,NY,,Eric Adams
Annapolis,MD,40812,Gavin Buckley
```

Write a Python program **or** a bash script that:

- Removes any lines where the state name is blank
- Changes any missing population values to 10000 (not a actually good idea in practice!)
- Sorts the rows by population, increasing
- Saves to a new file `goodcities.txt`

For the example above, the resulting file should be

goodcities.csv

```
city,state,population,mayor
Dansville,NY,4433,Dennis Mahus
New York,NY,10000,Eric Adams
Ithaca,NY,32108,Laura Lewis
Annapolis,MD,40812,Gavin Buckley
```

(Extra challenge: try doing this separately in Python *and* in bash)

3. Suppose there are 10 text files called `file1.txt`, `file2.txt`, ..., `file10.txt`.

   Write a **bash script** that displays each filename along with the number of lines in that file, in any order, like

```
390 file6.txt
3053 file10.txt
6124 file1.txt
1602 file9.txt
19416 file8.txt
2761 file5.txt
84 file2.txt
8511 file3.txt
17632 file4.txt
8675 file7.txt
```

   Your script should work **in parallel** so the lines in each file are counted simultaneously.

4. Now write a **Python program** that counts the number of lines in each of these 10 files `file1.txt`, `file2.txt`, etc. **in parallel** and prints only the name of the file that has the most lines.

   So for the files in the previous example, your program would just have one line of output:

```
file8.txt
```

5. Suppose there is a file `english.txt` that contains all English words in lowercase, like

`english.txt`

```
able
constable
brave
rave
revealing
stable
travesty
veal
```

Write a Python program that counts how many words appear within some other word.

In the case of the small dictionary above, the count would be 4, for able (const**able** or st**able**), rave (b**rave** or t**rave**sty), stable (const**able**), and veal (re**veal**ing).

Assume that the file has exactly 100,000 lines. For full credit, you should design your program run efficiently **in parallel**.

Consider the following small Python program:

prog.py

```
upto = int(input("Where to stop? "))
s = 0
n = 1
while n**2 < upto:
    s += n**2
    n += 1
with open('sum.txt', 'w') as fout:
    print(s, file=fout)
```

6. Describe how each of the major hardware components of the computer is used to execute the program above.

7. Suppose you run this program with the 'time' utility, and it reports that the "real" time was 10.3 seconds while the "user" time was 3.5 seconds. What could be causing that discrepancy?

8. Explain to your boss why adding more RAM to the computer won't make this program run any faster.

9. Explain something that could be done to make this program run faster, without purchasing any new hardware, and why that would work.

   (There are multiple correct answers! As a challenge, try to think of at least two solutions.)

10. Considering just this single line of the program:

    `s += n**2`

    what kind(s) of processor instructions would be needed to execute this line of Python code?

11. Imagine Batman and Robin are collaborating on a coding project using GitHub. They each have their own laptop and git directories.

    List all **five** places where a copy of their code is currently stored.

12. On Batman's laptop, imagine he creates a new file `cave.txt` in his git directory.

    What command(s) should he run to upload that file to GitHub?

13. In the meantime, Robin has edited a file `gotham.txt` that already existed in the git repository.

    What command(s) should Robin run to incorporate his edits with the new file from Batman, and sync everything back with GitHub?