

Name: _____

Problem	Points	Score
1	50	
2	50	
Total	100	
Extra Credit	25	
Extended Total	125	

Notes:

- (1) The first step in this exam is to create a workspace in the following directory:

`/data/courses/ece_1111/current/exams/ex_01`

Your directory should be your last name all lowercase, followed by an underscore, followed by your first name (e.g. “picone_joseph”). Set the permissions using “`chmod u+rwx,g-rwx,o-rwx <lastname>`” so only you have read and write permission to this directory. Create subdirectories within this directory: p01, p02, ... You will use these for problems 1 and 2, ... respectively. Put ALL your code in these directories. Do not touch your files after the exam is over.

- (2) For this exam you are allowed to open a terminal window on your computer, you are allowed to web surf with Google, but you cannot use online chat, ChatGPT or other interactive services. Your code must be your own original work.
- (3) You can only receive extra credit if you complete the first two problems correctly.

(50 pts) Problem No. 1:

In this directory:

```
/data/courses/ece_1111/current/exams/ex_01/picone_joseph
```

there are a bunch of text files distributed throughout a directory tree. Write a Python program, named *p01*, that locates all files ending in *.txt and adds up the number of lines that contain the letter sequence “aaa”. You should take a directory name and search string as a command line input (e.g., the above directory name), recurse through the directory tree, and examine all the *.txt files. Every time you find a pattern, you should print the filename and number of times that the pattern occurred in the file. After the last file is processed, you should print the sum.

Your output should look like this:

```
ece-000_[1]: p01 /data/courses/ece_1111/current/exams/ex_01/picone_joseph “aaa”
file: /data/courses/ece_1111/current/exams/ex_01/picone_joseph/f00.txt 1
file: /data/courses/ece_1111/current/exams/ex_01/picone_joseph/l01/f02.txt 6
...
the total number of lines containing “aaa” = 999
```

The value shown for the sum is not correct – it is just an example. The ordering of the files doesn’t matter, and the sum will be the same no matter in what order you process the files.

Use the per-file output to debug your program. Test it on one or two files before you attempt the entire data set.

(50 pts) Problem No. 2:

Write a one line Unix command that produces the same result as problem no. 1. Place this command in a directory *p02* using the filename *p02*. The interface should be the same:

```
p02 /data/courses/ece_1111/current/exams/ex_01/picone_joseph “aaa”
```

The output must show the sum exactly as shown above. You do not need to produce an output for each file.

(25 pts) Reflection:

You can only complete this problem if you complete the other two problems correctly.

Note that you now have two commands, *p01* and *p02*, that produce the same result. Whether these are Python programs, shell scripts, or C programs doesn’t really matter. From a user interface perspective, they function identically. Which program is faster? Explain.