

EE 577A Spring 2020
Pierluigi Nuzzo TA: Yinghua Hu

Lab 1 Part 1

Score:___/100

Student ID: _____

Name: _____

Assigned: Friday January 17th

Due: Tuesday January 28th, at 11:59pm. No late submissions are allowed.

In Lab1 Part 1 you can discuss your ideas, algorithms, etc. with your colleagues. However, you should submit your own implementation. You are **NOT** allowed to copy your code from others. This means you should not discuss or present your actual code. You can refer to online resources but be careful about citing them.

The objective of this assignment is to help you learn scripting, which is used in industry and academia for various simulation and automation steps of design and verification. You will use either Perl or Python for any EE577A scripting. Perl/Python scripting will be an important part of your final project and future lab assignments and may also appear as part of your exams.

In this lab, you are **NOT** allowed to use packages such as **re** (Regular Expression) in Python. If you do not know whether you are allowed to use a specific package, contact the TAs for confirmation.

1) (30 pts) Write a Python or Perl script (`lab1p1.py` or `lab1p1.pl`) that reads the sample input text file `text_in.txt`. The file may include special characters and numbers. Your script then writes every word in `text_in.txt` into an output file `text_out1.txt` with the following requirement: for each paragraph in `text_in.txt`, you should rewrite the words in the paragraph in **reverse order**. For example, if the text in `text_in.txt` is:

The University of Southern California is a private research university in Los Angeles, California. Founded in 1880, it is the oldest private research university in California. For the 2018-19 academic year, there were 20,000 students enrolled in four-year undergraduate programs. USC also has 27,500 graduate and professional students in a number of different programs, including business, law, engineering, social work, occupational therapy, pharmacy, and medicine. USC was one of the earliest nodes on ARPANET and is the birthplace of the Domain Name System. Other technologies invented at USC include DNA computing, dynamic programming, image compression, VoIP, and antivirus software.

then the text in `text_out1.txt` should be:

medicine. and pharmacy, therapy, occupational work, social engineering, law, business, including programs, different of number a in students professional and graduate 27,500 has also USC programs. undergraduate four-year in enrolled students 20,000 were there year, academic 2018-19 the For California. in university research private oldest the is it 1880, in Founded California. Angeles, Los in university research private a is California Southern of University The software. antivirus and VoIP, compression, image programming, dynamic computing, DNA include USC at invented technologies Other System. Name Domain the of birthplace the is

and ARPANET on nodes earliest the of one was USC

Notes:

- In this lab, a **word** is defined as a string of characters without any space in between.
- **Characters** can be letters, numbers, and special characters, such as “'”, “-”, “.”, and “,”. Examples of words include: four-year, 1880, law, and research.
- Space will only exist between two words OR at the end of a paragraph.
- You should take care of the line break such that no unnecessary spaces and line breaks will be added in your output file.

2) (30 pts) Analyze all the words, compute the word length distribution in the text file `text_in.txt`, and report the result in a new text file `text_out2.txt` by (strictly) following the format below.

| Word length | Count |
|-------------|-------|
| 1 | x |
| 2 | x |
| 3 | x |
| ... | ... |
| N | x |

Notes:

- Each of the two columns must be left-aligned.
- x is an integer representing the number of occurrences for the words with the corresponding word length.
- N is the maximum word length for the given text.

3) (40 pts) Encrypt each word of the text in `text_in.txt` with a simple text encryption scheme as follows. For each character in the word, find its ASCII code (you may use `ord()` in your Python script), denoted by $ASCII_{original}$. For example, the ASCII code of the character ‘a’ is 97, while the ASCII code of the character ‘3’ is 51. The ASCII code of the encoded character, denoted by $ASCII_{encoded}$, follows the formula below:

$$ASCII_{encoded} = [(ASCII_{original} - 33 + a) \bmod 94] + 33,$$

where a is equal to $[(\text{the last two digits of your USC student ID}) \bmod 94]$. Finally, output the encrypted text to a text file `text_out3.txt`. Examples are shown below:

Before encryption:

The University of Southern California is a private research university in Los Angeles, California.

After encryption:

l"} m(#0},-#.3)~ k)/. },{ ([y&#~), (#y #- y *,#0y.} ,}-}y,{ " /(#0},-#.3 #(d)- Y(!}&}-D [y&#~), (#yF

Submission Guidelines

Briefly explain your algorithms, ideas, comments in a pdf file. Name your pdf file based on

the following format: “yourfirstname_yourlastname_studentID_Lab1_Part1.pdf”.

Provide a readme.txt including any facts you think the grader should know to run and assess your work. This includes references: e.g.: in my Python code, lines 5-38, are indirectly (or mention, if directly) inspired by the code (lines numbers) in www.....

Submit your lab1p1.pl or lab1p1.py scripts as well as your generated text_out1.txt, text_out2.txt, and text_out3.txt files.

Submit through DEN->Assignments->Lab1Part1 Submission. Multiple submissions are allowed, and the latest submission will be graded.