ECE 473 Homework 1

Instructions on programming questions

You should modify the code in `hw1_submission.py` between
`# BEGIN_YOUR_CODE`
and
`# END_YOUR_CODE`
but you can add other helper functions outside this block if you want. Do not make changes to files other than `hw1_submission.py`.

Your code will be evaluated on two types of test cases, **basic** and **hidden**, which you can see in `hw1_grader.py`. Basic tests, which are fully provided to you, do not stress your code with large inputs or tricky corner cases. Hidden tests are more complex and do stress your code. The inputs of hidden tests are provided in `hw1_grader.py`, but the correct outputs are not. To run the tests, you will need to have `graderUtil.py` in the same directory as your code and `hw1_grader.py`. Then, you can run all the tests by typing

```
python hw1_grader.py
```

This will tell you only whether you passed the basic tests. On the hidden tests, the script will alert you if your code takes too long or crashes, but does not say whether you got the correct output. You can also run a single test (e.g., `3a-0-basic`) by typing

```
python hw1_grader.py 3a-0-basic
```

We strongly encourage you to read and understand the test cases, create your own test cases, and not just blindly run `hw1_grader.py`.

Problem 1: Programming preliminaries for the course

In this problem, you will implement a bunch of short functions. The main purpose of this exercise is to familiarize yourself with Python, but as a bonus, the functions that you will implement will come in handy in subsequent homeworks. Submit your code in the file `hw1_submission.py` on BrightSpace.

(a) Implement `findAlphabeticallyLastWord` in `hw1_submission.py`.

(b) Implement `euclideanDistance` in `hw1_submission.py`.

(c) Implement `mutateSentences` in `hw1_submission.py`.

(d) Implement `sparseVectorDotProduct` in `hw1_submission.py`.

(e) Implement `incrementSparseVector` in `hw1_submission.py`.

(f) Implement `findSingletonWords` in `hw1_submission.py`.

(g) Implement `computeLongestPalindromeLength` in `hw1_submission.py`.

Problem 2: Watch and comment on Codex demo

Watch the 30-minute video demo at the URL below (also listed on the course web page) for the OpenAI software called "Codex". Write a few paragraphs that include a quick summary of what Code is and your reaction. This reaction could include any of: what are the likely strengths and weaknesses of Codex? Is Codex likely to be useful for the kind of coding you may one day be employed to do? What are the downstream prospects for improvements on this kind of tool? What would you like to build with it? Are there any dangers in this kind of technology? Submit your paragraphs in typeset form in a file `hw1.pdf` on BrightSpace.

`https://www.youtube.com/watch?v=SGUCcjHTmGY`