Homework 2

Assigned: 9/14/2023

Due: 9/21/2023 at 11:59 PM

WARNING. Students may not work together. Students may discuss the problems with each other, but do not give any other student your solutions.

Install matplotlib into your python environment. In the slides for lecture 7 I showed how to do this for PyCharm. The slides are available in blackboard.

Place all files containing your answers to homework 2 in a directory named `hw2_last_first` where `last_first` is the student's last and first names separated by an underscore. For me, the directory would be `hw2_harrison_david`.

Place each answer in its own file or directory. When you are done your directory structure should look like.

```
$ ls -F
hw2_harrison_david/
$ cd hw2_harrison_david
$ ls -F
p1.txt  p2.txt  p3.txt  p4.txt  p5.txt  p6/  p7.py  p8.py
$ cd p6
$ ls
bag.py  test_bag.py
```

Each file contains the follwoing:

- p1.txt: question and answer to p1 from 3.10
- p2.txt: question and answer to p2 from 3.10
- p3.txt: question and answer to p3 from 3.10
- p4.txt: question and answer to p4 from 3.10
- p5: answer to question 5 below is written into bag.py and test_bag.py.
- p6.py: answer to question 6 given below that can be run in pycharm using matplotlib
- p7.py: answer to question 7 given below that can be run in pycharm using matplotlib

Zip or tar the directory `hw2_last_first` and submit them to blackboard in the same manner as was done for homework 1. On Mac OS or linux, it would look like this:

```
$ ls -F
hw2_harrison_david/
$ tar -czf hw2_harrison_david.tgz hw2_harrison_david
$ ls -F
hw2_harrison_david/ hw2_harrison_david.tgz
```

Submit `hw2_harrisond_david.tgz` to blackboard. If you are on windows, you may use zip, in which case the file submitted would be `hw2_harrison_david.zip`.

Problems 1-4 (1 point each) are the discussion questions in 3.10 of *Problem Solving with Algorithms and Data Structures using Python.*

Problem 5: (2 points) The Bag class from lectures 6 and 7 can be found in the class repository at

`https://git.cs.olemiss.edu/harrison/csci-356`

in

`lectures6and7/bag/bag.py`

Add an iterator class to Bag. The iterator class must pass the unit tests committed in the repository in the directory `hw2/bag/test_bag.py`. You will receive *partial credit if you do not write unit tests for your iterator class, or if the code lacks comments or type hints.* Write to test more conditions than the tests given in `test_bag.py`. The tests should cover edge conditions like the iterator should work as expected on an empty list.

Problem 6: (2 points) Write a program that verifies that the list index operator is O(1). The program must plot the run time of the list index operator as a function of n using matplotlib.

Problem 7: (2 points) Write a program that compares the performance of the `del` operator on lists and dictionaries. The main program should plot the run time of each on the same plot as a function of n. Also plot functions that bound the time complexity and print out what you think is the time complexity of `del` operators for lists and dictionaries using big-O notation. When measuring performance on the list `del` operator, be sure to delete items at random locations from the list.