MATH 4332

Assignment# 9

Due : 11/10/2023, Friday, before 11:59pm Term :Fall 2023

Important Note: Please use only materials covered so far to solve your problems. You will be given zero if the materials used are not yet covered.

---

1. You are given a file called `class_scores.txt`, where each line of the file contains a one word username and a test score separated by spaces, like below:.

   ```
   GWashington 83
   JAdams 86
   ```

   Write code that scans through the file, adds 5 points to each test score, and outputs the usernames and new test scores to a new file, `scores2.txt`.

2. You are given a file called `grades.txt`, where each line of the file contains a one word student username and three test scores separated by spaces, like below:.

   ```
   GWashington 83 77 54
   JAdams 86 69 90
   ```

   Write code that scans through the file and determines how many students passed all three tests.

3. You are given a file called `students.txt`. A typical line in the file looks like:

   ```
   walter melon            melon@email.msmary.edu              555-3141
   ```

   There is a name, an email address, and a phone number, each separated by tabs. Write a program that reads through the file line-by-line, and for each line, capitalizes the first letter of the first and last name and adds the area code `301` to the phone number. Your program should write this to a new file called `students2.txt`. Here is what the first line of the new file should look like:

   ```
   Walter Melon             melon@email.msmary.edu         301-555-3141
   ```

4. You are given a file called `baseball.txt`. A typical line of the file starts like below.

   ```
   Ichiro Suzuki     SEA     162     680     74     ...[more stats]
   ```

   Each entry is separated by a tab, `\t`. The first entry is the player's name and the second is their team. Following that are 16 statistics. Home runs are the seventh stat and stolen bases are the eleventh. Print out all the players who have at least 20 home runs and at least 20 stolen bases.

5. Ask the user to enter a string of lowercase letters. Then use `wordlist.txt` to find all the words that can be made from the letters of the user's string, taking into account frequencies. [Hint: Creating a dictionary of letter frequencies might help.]