CS103 – Fall 2025 – Lab08

General Instructions

1. In today's lab, you are required to complete all required exercises during class to receive attendance credit.
2. You are welcome to:
    - Work together with classmates.
    - Search online for help or documentation.
    - Ask the TA for guidance if you are stuck.
3. To receive credit, you must finish and show your solutions to the TA before leaving lab.
4. There are also extra (optional) exercises at the end for those who want to challenge themselves. These are not required for attendance credit but are recommended for practice.

Exercise Instructions

- Make a folder **Lab8** inside your **cs103fa25** folder.
- Create a new python file inside your Lab8 folder (lab08.py).

Required Exercises

EXERCISE 1:

Write a function "**`listCheckAndSort`**" that takes a list "**`L`**" and determines all even numbers where x >= -20 and x < 20. Your function will return a `tuple` containing two `lists`, the first sorted from lowest to highest and the second sorted highest to lowest. Assume the list "**`L`**" only includes integers. *HINT: A built-in function and its parameters will help you here.*

`Sample Input:`

```
L = [2,-16,6,77,100,-3,62,18,0,123,1,18,-18]
```

`Sample Output`

```
([-18, -16, 0, 2, 6, 18, 18], [18, 18, 6, 2, 0, -16, -18])
```

EXERCISE 2:

Write a function "**finalGrade**" that takes a list "**classList**" which contains a sublist for each student. Each sublist contains three grades: attendance, exam grade, and homework grade. You need to return a tuple that contains the average of attendance, average of exams, and average of homeworks for the whole class as `floats`.

Sample Input

```
classList = [[20,80,90], [10,60,40], [17, 77,56], [21,99,99]]
```

Sample Output

```
(17.0, 79.0, 71.25)
```

***Hints:** The class has 4 students

average attendance = (20+10+17+21)/4 = 17

average exams = (80+60+77+99)/4 = 79

average homeworks = (90+40+56+99)/4 = 71.25

EXERCISE 3:

Write a Python function **movieCompare** that takes two lists, **favorites1** and **favorites2**, representing two people's favorite movies. The function should:

- Display all the movies both people like (common movies).
- Display all the movies that are unique to each person (not liked by the other).

Sample Input

```
favorites1 = ["Inception", "Titanic", "Avatar", "The Matrix"]
favorites2 = ["Titanic", "The Dark Knight", "Inception", "Interstellar"]
```

Sample Output

```
Common movies: Inception, Titanic
Unique movies: Avatar, The Matrix, The Dark Knight, Interstellar
```

Extra (Optional) Challenges

These are not required for credit but will help you practice and strengthen your problem-solving skills.

Challenge 1: SimpleEncr(s,n)

Write a function **`SimpleEncr(s,n)`** function that takes a string and an integer value. Your function will apply a simple Caesar cipher to create a substitution cipher in which each letter of the plain text is substituted with a letter found by moving n places down the alphabet.

Hints: Check the lecture notes that we cover character and ascii code

Sample Input

```
abcd xyz
```

Sample Output

```
efgh bcd
```

Challenge 2: Find the Longest Word

Write a function **`longestWord(words)`** that takes a list of words and returns the longest word in the list.

If there are multiple words with the same length, return the first one.

Sample Input

```
words = ["apple", "banana", "kiwi", "strawberry", "pear"]
```

Sample Output

```
strawberry
```

Challenge 3: flattenList(L)

Write a function **`flattenList(L)`** that takes a list that may contain other lists and returns a **single flat list** containing all the elements.

Sample Input

```
L = [1, [2, 3], [4, [5, 6]], 7]
```

Sample Output

```
[1, 2, 3, 4, 5, 6, 7]
```

Challenge 4: Transpose a Matrix

Write a function **`transpose(matrix)`** that takes a nested list and returns its **transpose** (rows become columns).

Sample Input:

```
matrix = [[1, 2, 3],
          [4, 5, 6]]
```

Sample Output

```
[[1, 4],
 [2, 5],
 [3, 6]]
```

To get attendance credit, finish Exercises 1–3.

If you finish early, try the optional challenges!