**CS150 Worksheet 08 – Lists and List Comprehension**

Complete the following problems in Wing. Submit 3 .py files named: *lastname*Wk08P1.py, *lastname*Wk08P2.py, and *lastname*Wk08P3.py.

1. (5 pts) Write a program, using list comprehension, to accept a string as input from the user and return a string with vowels removed. Below is a function using a FOR-loop to remove vowels from a string. Rewrite this function to use list comprehension in your solution.

# using for loop to remove vowels

def removeVowels(sentence):

vowels = 'aeiou'

noVowels = [ ]

for ch in sentence:

if ch not in vowels:

noVowels.append(ch)

return ''.join(noVowels)

SAMPLE RUN (user input in bold)

Enter a sentence: **Today is the best day of my life.**

Tdy s th bst dy f my lf.

1. (5 pts) Write a program to fill a list with 10 random numbers between 1-10 (inclusive). Sort the list and then using list comprehension, create a squared list squaring all the even numbers in the list and create a cubed list cubing all the odd numbers in the list. Your output should display the original list, the sorted list, the even numbers squared list, and the odd numbers cubed list.

SAMPLE RUN 1 (your values will vary as they are random numbers)

Original list: [5, 6, 2, 1, 1, 2, 6, 4, 1, 4]

Sorted list: [1, 1, 1, 2, 2, 4, 4, 5, 6, 6]

Square of evens: [4, 4, 16, 16, 36, 36]

Cube of odds: [1, 1, 1, 125]

SAMPLE RUN 2 (your values will vary as they are random numbers)

Original list: [6, 10, 2, 3, 7, 7, 3, 3, 10, 3]

Sorted list: [2, 3, 3, 3, 3, 6, 7, 7, 10, 10]

Square of evens: [4, 36, 100, 100]

Cube of odds: [27, 27, 27, 27, 343, 343]

1. (10 pts) Write a program to determine your CS150 grade. Grades are weighted in categories as follows:
   1. Worksheets, Assignments: 25%
   2. Quizzes: 15%
   3. Written Exams: 20%
   4. Lab Exams: 20%
   5. Final Exam: 20%

Have the user input scores (may include float values) for each category separated by spaces (see sample runs below).

To calculate the grade:

1. average each category
2. take each category average times its weight (see above) to get weighted percentage
   1. if the final exam has not been taken yet, the user should enter a -1 and your program should not include the final exam in the grade
3. add the weighted percentages for all categories to get the overall grade
   1. final exam not included would be:

(assignAvg \* .25 + quizAvg \* .15 + writtenExAvg \* .20 + labExAvg \* .20) / .80

* 1. final exam included would be:

assignAvg \* .25 + quizAvg \* .15 + writtenExAvg \* .20 + labExAvg \* .20 + finalEx \* .20

1. determine the letter grade based on the overall grade as follows:

|  |  |
| --- | --- |
| Range | Letter |
| 93 – 100 | A |
| 90 – 92.99 | A- |
| 87 - 90.99 | B+ |
| 83 - 86.99 | B |
| 80 - 82.99 | B- |
| 77 - 79.99 | C+ |
| 73 - 76.99 | C |
| 70 – 72.99 | C- |
| 67 – 69.99 | D+ |
| 60 – 66.00 | D |
| 0 – 59.99 | F |

* 1. Display blank line between inputs and overall and letter grade output. Display overall grade with 2 decimal places (see sample run below)

Sample Run1 – no final exam yet (user input in bold)

Assignment scores (separated by a space): **100 90 80 70.5 60 50 80** **90.5**

Quiz scores: **95 75 80.5**

Written exam scores: **95 75**

Lab exam scores: **75.5 89.9**

Final exam score (enter -1 if not taken yet): **-1**

Overall grade: 81.84

Letter grade: B-

Sample Run 2 (user input in bold)

Assignment scores (separated by a space): **100 100 100 100 90 80 75** **100 100**

Quiz scores: **95 88 92**

Written exam scores: **80 88**

Lab exam scores: **100 85**

Final exam score (enter -1 if not taken yet): **90**

Overall grade: 90.52

Letter grade: A-