CIS 3260 – Introduction to Programming
Individual Assignment 8

Submission:

- Add your full name and email address as comments on top of each python program. Add appropriate comments at other locations of your python programs for readability.
- Name your python program as IA8Q#.py. For example, for question #1, your program should have a name IA8Q1.py
- For each question, put corresponding answers in a word document. Name your word document as **IA8.doc**
- Submit following items into iCollege -> Assessment -> Assignment 7
  - IA8.doc
  - IA8Q1.py
  - IA8Q2.py
  - IA8Q3.py

[Some Comments from your instructor]:
I highly recommend you use PyCharm to write following two programs. You can open existing python project and create two new python files IA8Q1.py, IA8Q2.py and IA8Q3.py.

1. [Objectives: 7.1,7.2,7.3,7.4,7.7]

1) [33 points] Assume you are given a task to analyze a list of ratings (integers, ranging from 1 to 5). The ratings are stored in a list as below. Please develop a program to ask user to enter a rating and then count the number of given ratings from the list.

```
ratings_list=[4,5,1,2,3,5,4,2,3,1,5,4,4,3]
```

```
Sample Run 1
Please enter a rating: 5
3 ratings from our data set are with value 5
```

```
Sample Run 2
Please enter a rating: 0
0 is not a valid rating
```

A. [5 points] System Analysis: What is(are) the input(s) and output(s)?
B. [5 points] System Design: List the steps to get output(s) from input(s) in English.[Hint: you can refer to the example in exercise 7.3]
C. [4 points] We have learned to two ways to traverse elements from list. Way 1 is for "read only" task (i.e., no **range** used in for loop header) ; way 2 is for "read and update" task (i.e., must use **range** in for loop header)). Which way you are going to use in this problem? And why you prefer to choose this way?
D. [5 points] Write your program and put your source code here.
E. [2 points] A screenshot of the output of program

2) Share with us TWO errors you experienced when writing this program. [It is common to experience errors. I will be surprised if you do not see any errors. In case this happens, you can make some errors by yourself.]

I . Error 1

A. [2 pts] Explain what is the type of the error (syntax, run-time, or logical error)?
B. [2 pts] How you resolve it?
C. [2 pts] How much time you spent to resolve it?

II. Error 2

A. [2 pts] Explain what is the type of the error (syntax, run-time, or logical error)?
B. [2 pts] How you resolve it?
C. [2 pts] How much time you spent to resolve it?

2. [Objectives: 7.1,7.2,7.3,7.4,7.6,7.7]

1) [33 points] Let us continue question1. But this time you are asked to write a program to count the frequency of each rating (1,2,3,4,5). Please use a list `counts_list` to store the counts. For example, count[0] stores the count of rating 0, count[1] stores the count of rating 1.

```
ratings_list=[4,5,1,2,3,5,4,2,3,1,5,4,4,3]
```

```
Sample Run
Rating 1: 2
Rating 2: 2
Rating 3: 3
Rating 4: 4
Rating 5: 3
```

A. [5 points] System Analysis: What is(are) the input(s) and output(s)
B. [4 points] What should be the length of list `counts_List`? How to create the list `counts_List` in Python?
C. [5 points] System Design: List the steps to get output(s) from input(s) in English.
D. [5 points] Write your program and put your source code here.
E. [2 points] A screenshot of the output of program.

2) Share with us TWO errors you experienced when writing this program.

I . Error 1

A. [2 pts] Explain what is the type of the error (syntax, run-time, or logical error)?
B. [2 pts] How you resolve it?
C. [2 pts] How much time you spent to resolve it ?

II. Error 2

A. [2 pts] Explain what is the type of the error (syntax, run-time, or logical error)?

B. [2 pts] How you resolve it?
C. [2 pts] How much time you spent to resolve it?

3. [Objectives: 7.1,7.2,7.3,7.4,7.6,7.7]
1)[34 points] It is still based on question 1. Assume you are given another list storing the name of users submitting the reviews. For example, "john" provided a review score 4; "alex" submitted a review score 1. Write a program to traverse the two lists and display the names of person(s) who gives the highest rating, and the person(s) who gives the lowest rating

```
ratings_list=[4,5,1,2,3,5,4,2,3,1,5,4,4,3]

names_list=["john", "alex", "anna", "max", "allen", "bria",
"david", "jane", "cathy","jecissa", "grace", "nick", "paul",
"hans",]
```

Sample Run

```
Highest rating is 5, given by alex,bria,grace
Lowest rating is 1, given by anna and jecissa
```

A. [5 points] System Analysis: What is(are) the input(s) and output(s)?
B. [5 points] System Design: List the steps to get output(s) from input(s) in English.
C. [4 points] We have learned to two ways to traverse elements from list. Way 1 is for "read only" task (i.e., no **range** used in for loop header) ; way 2 is for "read and update" task (i.e., must use **range** in for loop header)). Which way you are going to use in this problem? And why you prefer to choose this way?
D. [5 points] Write your program and put your source code here.
E. [3 points] A screenshot of the output of program

2) Share with us TWO errors you experienced when writing this program. [It is common to experience errors. I will be surprised if you do not see any errors. In case this happens, you can make some errors by yourself.]

I . Error 1
A. [2 pts] Explain what is the type of the error (syntax, run-time, or logical error)?
B. [2 pts] How you resolve it?
C. [2 pts] How much time you spent to resolve it?

II. Error 2
A. [2 pts] Explain what is the type of the error (syntax, run-time, or logical error)?
B. [2 pts] How you resolve it?
C. [2 pts] How much time you spent to resolve it?

Objectives:

Week 4

--- Chapter 3 ---

3.1 To write Boolean expressions using relational operators (§3.2).
3.2 To program with Boolean expressions (§3.3).
3.3 To implement selection control using one-way if statements (§3.4).
3.4 To implement selection control using two-way if-else statements (§3.5).
3.5 To implement selection control with nested if and multi-way if-elif-else statements (§3.6).
3.6 To combine conditions using logical operators (and, or, and not) (§3.10).
3.7 To use selection statements with combined conditions (§§3.11–3.12).

Week 7

--- Chapter 4 ---

4.1 To solve mathematics problems by using the functions in the math module (§4.2)
4.2 To represent and process strings and characters (§4.3).
4.3 To encode characters using ASCII and Unicode (§4.3.1).
4.4 To use the ord function to obtain a numerical code for a character and the chr function to convert a numerical code to a character (§4.3.2).
4.5 To represent special characters using the escape sequence (§4.3.3).
4.6 To test substrings using the in and not in operators (§4.3.8).
4.7 To compare strings (§4.3.9).
4.8 To use string functions min, max, and len (§4.3.10).

Week 8

--- Chapter 4 ---

4.9 To obtain a character in a string using the index operator [] (§4.3.11).
4.10 To obtain a substring in a string using the slicing operator [start : end] (§4.3.12).
4.11 Use repetition operator * to duplicate strings (§4.3.6).

4.12 To introduce objects and methods (§4.5).
4.13 To introduce the methods in the str class (§4.6).
4.14 To program using characters and strings (§4.7.1).
4.15 To invoke the print function with the end argument (§4.3.4).

--- Chapter 5 ---

5.1 To write programs for executing statements repeatedly using a while loop (§5.2).

Week 9

--- Chapter 5 ---

5.2 To control a loop with the user's confirmation and a sentinel value (§5.5).
5.3 To develop loops following the loop design strategy (§5.4).
5.4 To use for loops to implement counter-controlled loops (§5.6).
5.6 To implement program control with break. (§5.10).

Week 10

--- Chapter 6 ---

6.1 To understand what is a function (§6.1)
6.2 To define functions with formal parameters (§6.2).
6.3 To distinguish the differences between the functions that return and do not return a value (§6.4).
6.4 To invoke functions with actual parameters (i.e., arguments) (§6.3).
6.5 To determine the scope of variables (§6.8)

Week 11

--- Chapter 6 ---

6.6 To invoke a function using positional arguments or keyword arguments (§6.5)
6.7 To invoke functions defined from another program (§6.7)
6.8 To apply the concept of function in software development and design (§6.13)

Week 12

--- Chapter 7 ---
7.1 To describe why lists are useful in programming (§7.1).
7.2 To learn how to create lists (§7.2).
7.3 To use the len, min, max, sum, and random.shuffle functions with a list (§7.2.2).
7.4 To access list elements by using indexed variables (§7.2.3).
7.5 To obtain a sublist from a larger list by using the slicing operator [start : end : step] (§7.2.4).
7.6 To use the + (concatenation), * (repetition), and in/not in operator
7.7 To traverse elements in a list using a for loop (§7.2.6).