ENGG1330 Computer Programming

Tutorial 6

Objectives:

1. Demonstrate working knowledge of array
2. Solve problems with array

Tutorial participation (1%):

- t6_vpl_1
- Submission period: **within your OWN tutorial period**

Tutorial/take-home exercises (2%):

- Remaining problems in the worksheet
- Submission deadline: **noon, 22-MAR-2023 (Wednesday)**

t6_vpl_1. Two different words are a "reverse pair" if each is the reverse of the other. Write a program that asks user to enter a sentence and lists all the reverse pairs in table format (column width is 10 characters), assuming that all alphabets are in lowercase in the sentence with no punctuation mark.

Hints:

- You can use the method `split()` to break a string into a list of words.

Example:

```
>>> sentence='Department of Electrical and Electronic Engineering'
>>> words=sentence.split()
>>> words
['Department', 'of', 'Electrical', 'and', 'Electronic', 'Engineering']
```

- A string can be reversed as follows.

Example:

```
>>> word='desserts'
>>> word[::-1]
'stressed'
```

Sample cases and screenshots

```
========== RESTART: /Users/csvlee/Documents/1330/lab/tut6/tut6_3.py ==========
Enter a sentence: she feels stressed after eating a lot of desserts
stressed  desserts
desserts  stressed
>>>
========== RESTART: /Users/csvlee/Documents/1330/lab/tut6/tut6_3.py ==========
Enter a sentence: did you stab the bats inside the cave
stab      bats
bats      stab
>>>
========== RESTART: /Users/csvlee/Documents/1330/lab/tut6/tut6_3.py ==========
Enter a sentence: this was the smart phone I saw when the trams are leaving
was       saw
smart     trams
saw       was
trams     smart
```

column width: 10 characters, left-aligned

t6_vpl_2. Write a program to map marks to grades using the following mapping table.

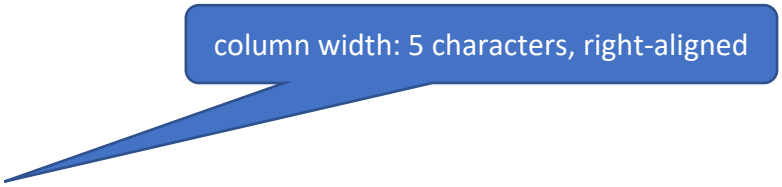
Mark	Grade
85 or above	A
75 to 84	B
65 to 74	C
50 to 64	D
Below 50	F

The program asks user to enter number of students and an integral mark for each student in the range of [0, 100]. After all marks are read, the program outputs the marks and grades in table format (column width is 5 characters).

Sample cases and screenshots

```
============ RESTART: /Users/csvlee/Documents/1330/lab/tut6/tut2.py ============
Number of students? 5
Student 1: 71
Student 2: 51
Student 3: 84
Student 4: 99
Student 5: 16
   71   51   84   99   16
    C    D    B    A    F

============ RESTART: /Users/csvlee/Documents/1330/lab/tut6/tut2.py ============
Number of students? 10
Student 1: 41
Student 2: 65
Student 3: 78
Student 4: 56
Student 5: 100
Student 6: 89
Student 7: 0
Student 8: 91
Student 9: 71
Student 10: 82
   41   65   78   56  100   89    0   91   71   82
    F    C    B    D    A    A    F    A    C    B
```



t6_vpl_3. Write a program to do the following tasks.

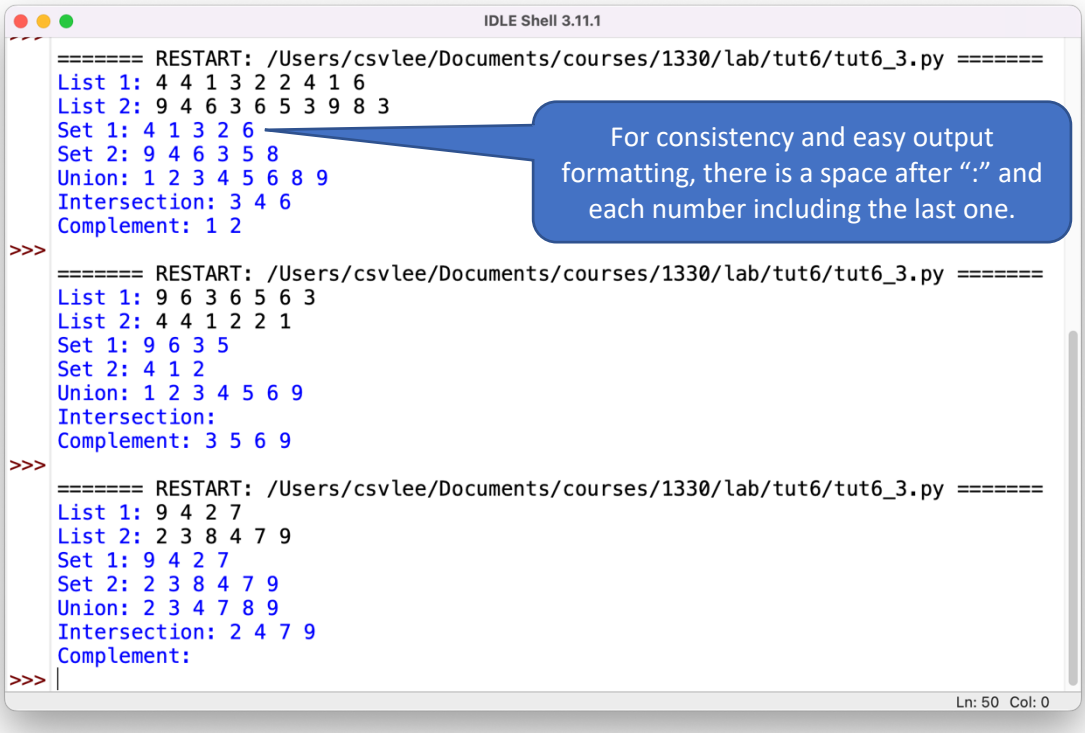
- **a.** Reads two lists of integers from the user who enters a list in a line.
- **b.** Removes all duplicate in the two lists, after which they are called sets.
- **c.** Outputs the two sets of integers in original order.
- **d.** Finds and outputs the union of the two sets in ascending order.
- **e.** Finds and outputs the intersection of the two sets in ascending order.
- **f.** Finds and outputs the complement of the two sets in ascending order.

Except for `split()` and methods on list, **using other methods is not allowed**.

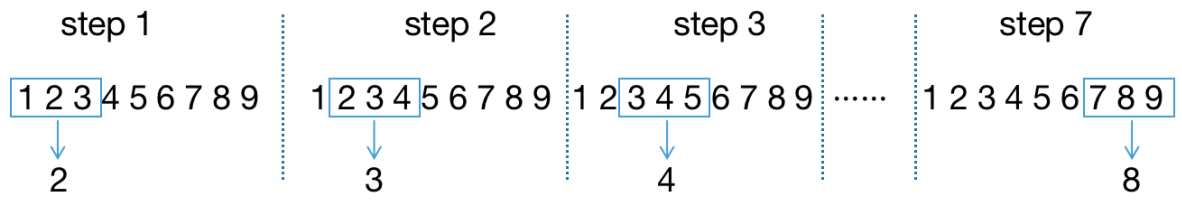
Hints:

- The union of A and B is the set of all numbers that are members of A or of B or of both.
- The intersection of A and B is the set of all numbers that are members of both A and B.
- The relative complement of B in A (A − B) is the set of all numbers that are members of A, but not members of B.
- For details, check wiki.

Sample cases and screenshots



t6_vpl_4. Suppose there is a window with a size of *k* (*k* > 2) sliding on an array of numbers in the direction of left to right. In each step, the average value of the *k* numbers in the window is calculated and appended to an output array. For example, if the length of array is 9 and the window size is 3, the window is sliding on the array as follows.



Write a program to implement this task. The program first asks user to input an array of numbers and then asks user to input the window size. After sliding the window on the array of numbers, the program displays the contents of the output array. Note the following additional information / requirements.

- If the length of input array is *n* and the window size is *k*, the number of average values in the output array should be *n* - *k* + 1. You can derive it by yourself.
- The input numbers are positive integers entered in a line, separated by a white space.
- The average values in the output array are displayed as floating-point numbers with 2 digits after the decimal point, separated by a white space.
- If the length of input array is less than the window size, the output array is empty.

Except for `split()` and methods on list, **using other methods is not allowed**.

Sample cases and screenshots