

## 111A Introduction to Computer and Computer Science

### Homework Assignment #7

**Due: 11/21 12:00:00**

In this homework assignment, you will practice how to apply the concept of object-oriented programming to solve a practical issue. As you have learned all the related knowledge during the lectures, you should have the ability to accomplish this.

#### Problem #1: Numerals

Numerals are one of the most important symbols in the history of mankind. They have been widely used in many parts of your life such as currency systems, mathematics, religions, .... About 2000 years ago, the Roman Empire had their numeral system to represent numbers. Unlike the current numeral system based on Arabic numerals, the system of Roman numerals uses special characters:

Symbol	Value
I	1
V	5
X	10
L	50
C	100
D	500
M	1000

In this homework, we are going to develop a program, which aims to transform an arbitrary Arabic numeral into Roman numeral. For simplicity, we only consider positive integers less than 4000 in this homework. In addition, you have to create some rules to tell Python how to deal with the operations like addition, subtraction and comparison of Roman numerals. You will need to use “class” to accomplish this task.

Please create a *class object* called **numerals** which can take **Arabic (int)** or **Roman numerals (str)** as input and meet the specification as shown below:

1. Show **Roman numerals** when you *print*.
2. Follow the **Roman numerals** rules:
  - (1) If one or more symbols are placed after another letter of greater value, add amount, vice versa.

- (2) The value of the symbol is added to itself, as many times as it is repeated.  
Ex. II = 2, III = 3.
- (3) Symbols V, L, D will never repeat.  
Ex. XV = 15, not VVV.
- (4) The maximum repeated times of a symbol is three.  
Ex. IV = 4, not IIII.
- (5) Subtraction only uses symbols I, X and C.  
Ex. XLV = 45, not VL.
- (6) Subtraction only uses symbol nearby.  
Ex. XCIX = 99 ... (100 - 10 + 10 - 1), not IC...(100 - 1).
- 3. It can be compared (>, <, ≤, ≥, =).
- 4. It can be operated (+, -, ×).
- 5. Show the corresponding message if the number is less equal than 0 or greater equal than 4000.

Test cases:

```

1. print(numerals('IV'))
   IV
2. print(numerals(4))
   IV
3. print(numerals(4300))
   Number out of range
4. print(numerals('DDD'))
   Wrong Roman numerals formation
5. print(numerals('IV')+numerals(5))
   IX
6. print(numerals('IV') <= numerals('V'))
   True
    
```

### Hand in procedure:

As we had mentioned in the lecture, you should list all your collaborators in your programs. Here is the template:

```
""  
Created on Sun Aug 7 01:23:45 2022  
  
@author: Xi Winnie, student ID  
  
@collaborators: Jane Doe, her student ID  
                John Doe, his student ID  
""
```

Please save your code as a “.py” file, where the file name should follow this format:

111A\_hw#7\_ID.py

For example,

111A\_hw#7\_0123456789.py

Please be aware. **We are not going to accept any homework file with wrong file name or without signature.** Please double check the content of your file.

Once you have accomplished your works, you can upload your homework to the “E3@NYCU” system. There will be a section for uploading your homework.