Assignment 1

COMP9021, Term 3, 2019

1. GENERAL MATTERS

1.1. Aim. The purpose of the assignment is to:

- develop your problem solving skills;
- design and implement the solution to a program in the form of a medium sized Python program;
- practice the use of arithmetic computations, tests, repetitions, lists, and strings.

1.2. Submission. Your program will be stored in a file named `roman_arabic.py`. After you have developed and tested your program, upload it using WebCMS. Assignments can be submitted more than once; the last version is marked. Your assignment is due by **Sunday 27 October 2019, 22:00 (Week 6)**.

1.3. Assessment. The assignment is worth 10 marks. It is going to be tested against a number of inputs. For each test, the automarking script will let your program run for 30 seconds.

Late assignments will be penalised: the mark for a late submission will be the minimum of the awarded mark and 10 minus the number of full and partial days that have elapsed from the due date.

The outputs of your programs should be **exactly** as indicated.

1.4. Reminder on plagiarism policy. You are permitted, indeed encouraged, to discuss ways to solve the assignment with other people. Such discussions must be in terms of algorithms, not code. But you must implement the solution on your own. Submissions are routinely scanned for similarities that occur when students copy and modify other people's work, or work very closely together on a single implementation. Severe penalties apply.

2. DESCRIPTION

You will design and implement a program that prompts the user for an input with `How can I help you?` . User input should be one of three possible kinds:

- `Please convert ***`
- `Please convert *** using ***`
- `Please convert *** minimally`

If user input is not of this form, with any occurrence of `***` an arbitrary nonempty sequence of nonspace symbols, then the program should print out **Whatever mix of symbols? $ * parenthesis, brackets are accepted?**

<div align="center">

`I don't get what you want, sorry mate!`

</div>

and stop.

2.1. First kind of input. In case the user inputs `Please convert ***`, then `***` should be either a strictly positive integer (whose representation should not start with 0) that can be converted to a Roman number (hence be at most equal to 3999), or a valid Roman number; otherwise, the program should print out **Use Uppercase**

<div align="center">

`Hey, ask me something that's not impossible to do!`

</div>

and stop. If the input is as expected, then the program should perform the conversion, from Arabic to Roman or from Roman to Arabic, and print out the result in the form

<div align="center">

`Sure! It is ***`

</div>

2.2. **Second kind of input.** In case the user inputs `Please convert *** using ***`, then the first `***` should be a strictly positive integer (whose representation should not start with 0) or a sequence of (lowercase or uppercase) letters and the second `***` should be a sequence of *distinct* (lowercase or uppercase) letters. Moreover,

- the second `***` is intended to represent a sequence of so-called *generalised Roman symbols*, the classical Roman symbols corresponding to the sequence `MDCLXVI`, whose rightmost element is meant to represent 1, the second rightmost element, 5, the third rightmost element, 10, etc.;
- if it is not an integer, the first `***` is intended to represent a so-called *generalised Roman number*, that is, a sequence of generalised Roman symbols that can be decoded using the provided sequence of generalised Roman symbols similarly to the way Roman numbers are represented.

If that is not the case, of if it is not possible to convert the first `***` from Arabic to generalised Roman or from generalised Roman to Arabic, then the program should print out

<div align="center">

`Hey, ask me something that's not impossible to do!`

</div>

and stop. If the input is as expected and the conversion can be performed, then the program should indeed perform the conversion, from Arabic to generalised Roman or from generalised Roman to Arabic, and print out the result in the form

<div align="center">

`Sure! It is ***`

</div>

2.3. **Third kind of input.** In case the user inputs `Please convert *** minimally`, then `***` should be a sequence of (lowercase or uppercase) letters. The program will try and view `***` as a generalised Roman number with respect to some sequence of generalised Roman symbols. If that is not possible, then the program should print out

<div align="center">

`Hey, ask me something that's not impossible to do!`

</div>

and stop. Otherwise, the program should find the *smallest* integer that could be converted from `***`, viewed as some generalised Roman number, to Arabic, and output a message of the form

<div align="center">

Sure! It is `***` using `***`

</div>

2.4. **Sample outputs.** Here are a few tests together with the expected outputs. The outputs of your program should be exactly as shown.

```
$ python3 roman_arabic.py
How can I help you? Please do my assignment...
I don't get what you want, sorry mate!
$ python3 roman_arabic.py
How can I help you? please convert 35
I don't get what you want, sorry mate!
$ python3 roman_arabic.py
How can I help you? Please convert 035
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert 4000
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert IIII
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert IXI
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
```

How can I help you? Please convert 35
Sure! It is XXXV
$ python3 roman_arabic.py
How can I help you? Please convert 1982
Sure! It is MCMLXXXII
$ python3 roman_arabic.py
How can I help you? Please convert 3007
Sure! It is MMMVII
$ python3 roman_arabic.py
How can I help you? Please convert MCMLXXXII
Sure! It is 1982
$ python3 roman_arabic.py
How can I help you? Please convert MMMVII
Sure! It is 3007
$ python3 roman_arabic.py
How can I help you? Please convert 123 by using ABC
I don't get what you want, sorry mate!
$ python3 roman_arabic.py
How can I help you? Please convert 123 ussing ABC
I don't get what you want, sorry mate!
$ python3 roman_arabic.py
How can I help you? Please convert XXXVI using VI
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert XXXVI using IVX   **Why not? because it doesnt follow the roman numbers rules?**
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert XXXVI using XWVI   **Why not? Dont follow the rules**
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert I using II
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert _ using _
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert XXXVI using XVI
Sure! It is 36
$ python3 roman_arabic.py
How can I help you? Please convert XXXVI using XABVI
Sure! It is 306
$ python3 roman_arabic.py
How can I help you? Please convert EeDEBBBaA using fFeEdDcCbBaA
Sure! It is 49036
$ python3 roman_arabic.py
How can I help you? Please convert 49036 using fFeEdDcCbBaA
Sure! It is EeDEBBBaA
$ python3 roman_arabic.py
How can I help you? Please convert 899999999999 using AaBbCcDdEeFfGgHhIiJjKkLl
Sure! It is Aaaabacbdcedfegfhgihjikjlk
$ python3 roman_arabic.py   **Roman numbers are based JUST in 7 symbols**
How can I help you? Please convert ABCDEFGHIJKLMNOPQRST using AbBcCdDeEfFgGhHiIjJkKlLmMnNoOpPqQrRsStT
Sure! It is 11111111111111111111
$ python3 roman_arabic.py   **There is a difference between uppercase and lowcase**

How can I help you? Please convert 1900604 using LAQMPVXYZIRSGN
Sure! It is AMAZING
$ python3 roman_arabic.py
How can I help you? Please convert ABCD minimally using ABCDE
I don't get what you want, sorry mate!
$ python3 roman_arabic.py
How can I help you? Please convert ABCD minimaly
I don't get what you want, sorry mate!
$ python3 roman_arabic.py
How can I help you? Please convert 0I minimally
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert ABAA minimally
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert ABCDEFA minimally
Hey, ask me something that's not impossible to do!
$ python3 roman_arabic.py
How can I help you? Please convert MDCCLXXXVII minimally
Sure! It is 1787 using MDCLXVI
$ python3 roman_arabic.py
How can I help you? Please convert MDCCLXXXIX minimally
Sure! It is 1789 using MDCLX_I
$ python3 roman_arabic.py
How can I help you? Please convert MMMVII minimally
Sure! It is 37 using MVI
$ python3 roman_arabic.py
How can I help you? Please convert VI minimally
Sure! It is 4 using IV
$ python3 roman_arabic.py
How can I help you? Please convert ABCADDEFGF minimally
Sure! It is 49269 using BA_C_DEF_G
$ python3 roman_arabic.py
How can I help you? Please convert ABCCDED minimally
Sure! It is 1719 using ABC_D_E