

COMP9021 PRINCIPLES OF PROGRAMMING

Term 3, 2022

Assignment 1

1. General matters

1.1. Aim

The purpose of this 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**
- use **procedural** programming.

1.2. Submission

Your program will be stored in a file named [roman_arabic.py](#). Assignments can be submitted more than once. The last version just before the due date and time will be marked.

Your assignment is due **Monday 24 October 2022 @ 10:00am Sydney time (Week 7)**.

1.3. Assessment

The assignment is worth **14 marks**. It will be tested against a number of inputs. For each test, the automarking script will let your program run for **30 seconds**. The outputs of your program should be **exactly** as indicated.

Late submission with **5% penalty per day** is allowed up to **5 days** from the due date, that is, any late submission after **Saturday 29 October Week 7 @ 10am (Sydney time)** will be discarded.

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 the user input is not of this form, with any occurrence of `***` an arbitrary **nonempty** sequence of **non-space symbols**, then the program should print out:

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

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:

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

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:

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

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, or 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:

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

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:

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

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:

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

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

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

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
```

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

```
$ python3 roman_arabic.py
```

```
How can I help you? Please convert XXXVI using XWVI
```

```
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
```

```
How can I help you? Please convert ABCDEFGHIJKLMNOPQRST using  
AbBcCdDeEfFgGhHiIjJkKlLmMnNoOpPqQrRsStT
```

```
Sure! It is 11111111111111111111
```

```
$ python3 roman_arabic.py
```

```
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 minimally
```

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

```
$ python3 roman_arabic.py
```

```
How can I help you? Please convert OI 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
```