

國立清華大學資訊工程學系
11210 CS 410000 計算機結構
Homework 2

Deadline: 2023/10/16 (23:59)

There are two parts in this homework.

PART I. (Load, Store, Add, Sub)

Please load the data 4(\$gp) as A, 8(\$gp) as B, and 12(\$gp) as C, and do the following calculations.

$D = A - C + B$, store D to 16(\$gp)

$E = B + C - 15$, store E to 4(\$gp)

global pointer(gp): A system defined pointer which is used to refer to constant.

Hint

We will give a template called arch_hw2_p1_template.asm, just open it using Mars4_5.jar, write your code within the ##### block in the file (i.e., line 36~41), but DO NOT modify the code elsewhere. Please refer to the following figure.

```
# =====  
  
# The following block helps you practice with the R-type instructions,  
# including ADD and SUB, and also LOAD and STORE.  
# Here, please read data from 4($gp), 8($gp) and 12($gp),  
# store A - C + B to 16($gp), and  
# store B + C - 15 to 4($gp)  
#####  
# @@@ write the code here
```

After you write your code, save it. Next, press “F3” to assemble the code.

(Make sure there is no error!) Next, press “F5” to run.

The “Run I/O” screen should show the result like the following figure.

Mars Messages	Run I/O
	This is the first part of Homework 2. {Load, Store, Add, Sub} A = 48 B = 19 C = 123 A - C + B = -56 B + C - 15 = 127 -- program is finished running --
Clear	

PART II. (Branch Loop. System call. Arithmetic Operations)

Please complete the MIPS assembly code according to the problem described.

Description:

Roman numerals are represented by seven different symbols: **I, V, X, L, C, D and M.**

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

For example, 2 is written as II in Roman numeral, just two one's added together. 12 is written as XII, which is simply X + II. The number 27 is written as XXVII, which is XX + V + II.

Roman numerals are usually written largest to smallest from left to right. However, the numeral for four is not IIII. Instead, the number four is written as IV. Because the one is before the five we subtract it making four. The same principle applies to the number nine, which is written as IX. There are six instances where subtraction is used:

- I can be placed before V (5) and X (10) to make 4 and 9.
- X can be placed before L (50) and C (100) to make 40 and 90.
- C can be placed before D (500) and M (1000) to make 400 and 900.

Given an integer, convert it to a roman numeral.

You must continuously input numbers for them to be converted to Roman numerals until you input 0, at which point the program will stop.

Example 1:

Input: Please enter a number(Input 0 to exit): 3

Output: The Roman is: III

Explanation: 3 is represented as 3 ones.

Example 2:

Input: Please enter a number(Input 0 to exit): 58

Output: The Roman is: LVIII

Explanation: L = 50, V = 5, III = 3.

Example 3:

Input: Please enter a number(Input 0 to exit): 1994

Output: The Roman is: MCMXCIV

Explanation: M = 1000, CM = 900, XC = 90 and IV = 4.

Constraints:

- $1 \leq \text{num} \leq 3999$

Format:

To facilitate the verification of your program correctness, **please format the output according to the specified format**, as explained in the following example.

Example:

```
Please enter a number(Input 0 to exit): 3
The Roman is: III
Please enter a number(Input 0 to exit): 58
The Roman is: LVIII
Please enter a number(Input 0 to exit): 1994
The Roman is: MCMXCIV
Please enter a number(Input 0 to exit): 0
bye
-- program is finished running --
```

Do not input on the next line.

Please use uppercase for the output Roman numerals

After you input 0, output "bye" and do not add a new line after outputting "bye".

← This line is automatically generated after your program exits and does not need to be manually output.

Hint

- This is a problem on LeetCode, and it is recommended that you first validate the correctness of your code on the platform before transferring it into MIPS code.

Reference: [Integer to Roman - LeetCode](#)

Note: There are some differences in the requirements of the problem compared to LeetCode.

- TA will use different numbers (within the constraint range) to test the correctness of your program.

Submission (Two assembly programs)

Please name your assembly program with your student ID; for example, arch_hw2_p1_102062801.asm & arch_hw2_p2_102062801.asm, and upload these 2 files onto eeclass.

Grading Criteria

Correctness: 80%

Comments in your code: 10%

Output format: 10%

嚴格禁止抄襲，抄襲者與被抄襲者一律 0 分！

MARS (MIPS Assembler and Runtime Simulator)

1. MARS can assemble and simulate the execution of MIPS assembly language programs. Please refer to the following URL to download Mars4_5.jar:

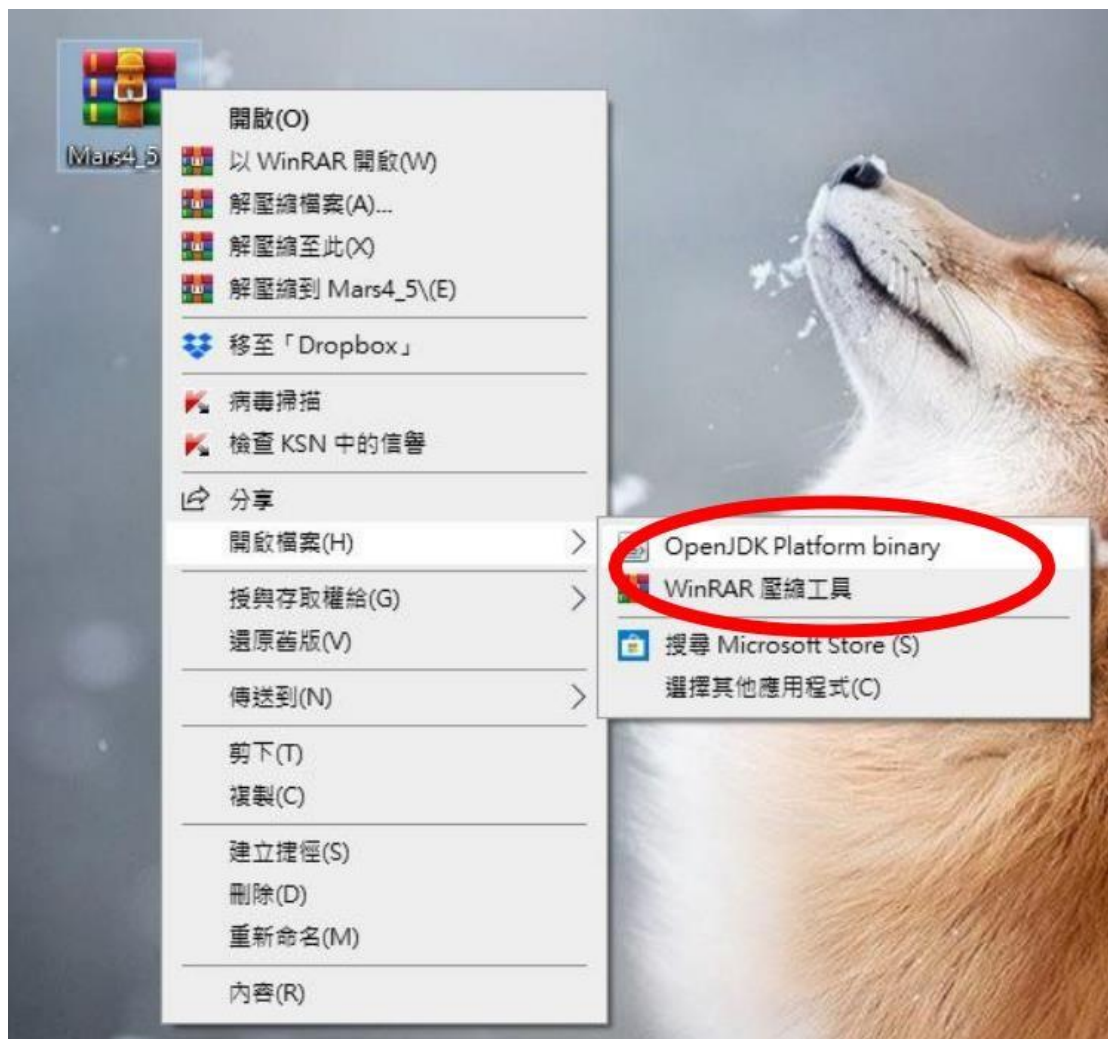
<http://courses.missouristate.edu/kenvollmar/mars/download.htm>

Download MARS **V4.5, Aug. 2014** (jar archive including Java source code)

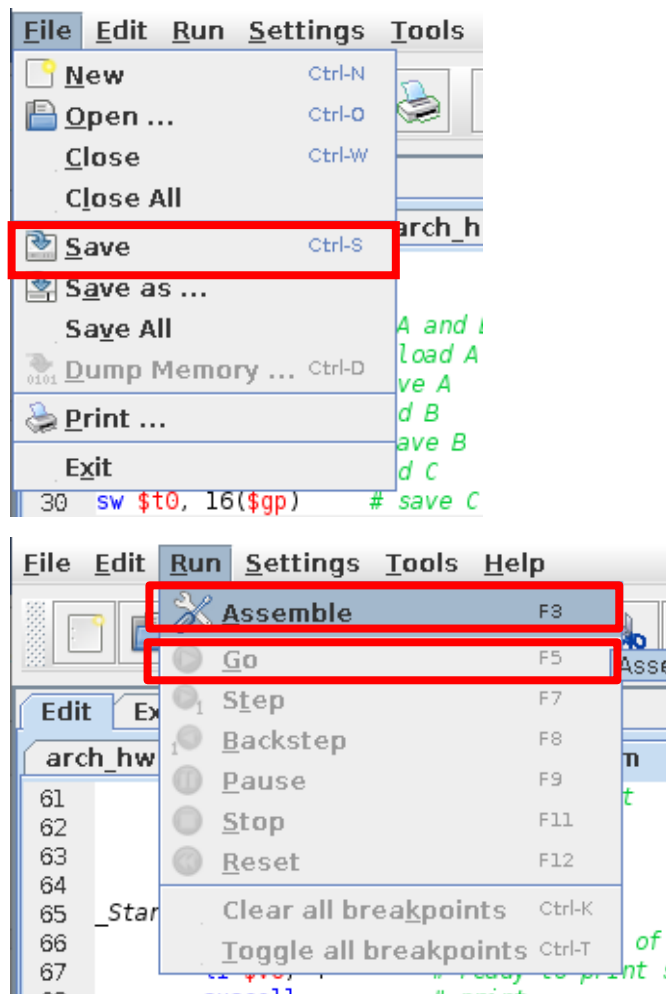
2. MARS is developed with Java language, and it requires JRE (Java Runtime Environment) installed on your computer. Please refer to the following URL to download JRE:

<https://www.oracle.com/java/technologies/downloads/>

3. After you download the MARS, it is a “.jar” file. Please DO NOT decompress it. You can open the MARS by following method.



4. Usage of MARS: P. S. Save your file, Assemble and Go



After saving the file, you can use Assemble button / F3 to assemble your code and use Go button / F5 to execute it.

The result will be shown in the Run I/O window.

5. If you want to execute Mars on CAD server, please use the option -X to run the graphic application.

ssh -X ic56

java -jar Mars4_5.jar

Appendix

1. This is a reference for the usage of syscall, you can learn how to do printf, scanf and generate random number in MIPS from this.
<https://courses.missouristate.edu/kenvollmar/mars/help/syscallhelp.html>
2. Two references for finding the functionality of MIPS instruction.
 - a. <http://alumni.cs.ucr.edu/~vladimir/cs161/mips.html>
 - b. [mips指令简单入门_mips冒号是什么意思](#)
[_Follow My Heart的博客-CSDN博客](#)