sIntroduction to Computer Organization (CCIT4026)

HKU SPACE Community College

(Lab 03 – CL01/CL07/CL08)

Instructions:

1. Fill in your personal information on the provided answer sheet (WORD) file.
2. The lab. submission consists of two components: a) A lab. report in PDF format which includes the material stated in the lab. sheet and b) The original assembly source code file for output verification.
3. Submission missing **EITHER** the lab. report (.pdf) **OR** the final source code (.asm) would **NOT** be graded.
4. Screen captures your final assembly source codes (with colors) and the input/output messages onto a WORD file and then convert to PDF format. The file would be considered as lab. report.
5. **Source code listing shown in the lab. report file CANNOT be superseded by the source code ".asm" file. Mark deduction would be imposed if no source code listing is shown in the report file.**
6. Upload **BOTH** the lab. report PDF file **AND** the final source code to SOUL.
7. Make sure the source codes are readable and neatly formatted. Unclear, incorrectly-sized or codes of poor readability would NOT be graded.
8. Each instruction should be associated with suitable comments to improve readability.
9. All problems are expected to be answered by using the ***instructions*** mentioned in the course materials ONLY. If you have to bring in instructions out of this scope, **prior consent** from the lecturer should be sought and a proper citation has to be provided otherwise the answers would be considered as INCORRECT.
10. Make sure to run (and single-step) your codes on MARS simulator to confirm its functionality before submission.
11. **Photo capture of the results** shown on the monitor is **NOT** acceptable. Use screen capturing software such as "Snipping Tool" on Window, then copy and paste the images onto the WORD file.
12. **No late submission will be accepted.**
13. Please read the Student Handbook for issues related to academic honesty. The College places very high importance on honesty in academic work submitted by students, and adopts a policy of zero tolerance on cheating and plagiarism. Any related offense may be passed to the Guidance Committee. Possible disciplinary action, including termination of study, may follow.
14. By making a submission, you also confirm to have read and understood the above-listed instructions and HKU SPACE Community College's academic honesty policy, and agree to comply with them. Failure to comply with the above instructions may lead to disqualification of your assignment results and possible disciplinary actions.

Introduction

In this lab, you are required to write a simple calculator to perform addition, subtraction, multiplication and division.

Lab Tasks

In this Lab., you are required to write a MIPS program to make use of the addition, subtraction, logical bit shift, multiplication and division instructions to compute the results based on the following equation,

$$Result = (quotient{:}remainder) = \frac{((B + C) * A - (D \ll 4))}{E}$$

where *A*, *B*, *C*, *D* and *E* are five integers and satisfied with the following inequality,

$$5 \leq E < D < C < B < A \leq 20$$

The program should ask the user to enter these five integers in a specific order and the following intermediate results are stored to the following variables defined in the data memory:

$$var1 = (B + C)$$
$$var2 = var1 * A$$
$$var3 = var2 - (D \ll 4)$$
$$quotient = var3 \,/ E \quad \text{(The quotient of the } Result\text{)}$$
$$remainder = var3 \,\%\, E \quad \text{(The remainder of the } Result\text{)}$$

(Note: Only consider the lower 32-bit of the multiplication result)

The following variables should be defined in your program,

```
17  E:              .space 4        # input E
18  D:              .space 4        # input D
19  C:              .space 4        # input C
20  B:              .space 4        # input B
21  A:              .space 4        # input A
22
23  # location to store intermediate results
24  var1:           .space 4        #
25  var2:           .space 4        #
26  var3:           .space 4        #
27  quotient:       .space 4        #
28  remainder:      .space 4        #
```

Below is the Pseudo-code of this program:

Step 1:

Step 1.1 # Display a prompt, “Please enter the value E: ”

Step 1.2 # Setup to wait for the user to enter a non-zero positive integer.

Step 1.3 # Store the entered value to the variable *E* defined in the data segment.

Step 2:

Step 2.1 # Display a prompt, “Please enter the value *D*: ”

Step 2.2 # Setup to wait for the user to enter a non-zero positive integer

Step 2.3 # Store the entered value to the variable *D* defined in the data segment.

Step 3:

Step 3.1 # Display a prompt, “Please enter the value *C*: ”

Step 3.2 # Setup to wait for the user to enter a non-zero positive integer

Step 3.3 # Store the entered value to the variable *C* defined in the data segment.

Step 4:

Step 4.1 # Display a prompt, “Please enter the value *B*: ”

Step 4.2 # Setup to wait for the user to enter a non-zero positive integer

Step 4.3 # Store the entered value to the variable *B* defined in the data segment.

Step 5:

Step 5.1 # Display a prompt, “Please enter the value *A*: ”

Step 5.2 # Setup to wait for the user to enter a non-zero positive integer

Step 5.3 # Store the entered value to the variable *A* defined in the data segment.

Step 6:

Step 6.1 # Load the value stored in variable *B* and *C* to two registers.

Step 6.2 # Add the values in the two registers.

Step 6.3 # Store the addition result to the variable *var1*.

Step 7:

Step 7.1 # Load the value stored in variable *A* to a register.

Step 7.2 # Multiply the value in *var1* to the register.

Step 7.3 # Store the lower 32-bit of the multiplication result to the variable *var2*.

Step 8:

Step 8.1 # Load the values stored in variable *D* to a register.

Step 8.2 # Left shift the register value.

Step 8.2 # Subtract the bit shifted result from *var2*.

Step 8.3 # Store the subtraction result to the variable *var3*.

Step 9:

Step 9.1 # Load the value stored in variable *E* to a register.

Step 9.2 # Divide the *var3* in the register by the variable *E*.

Step 9.3 # Store the quotient of the division result to the variable *quotient*.

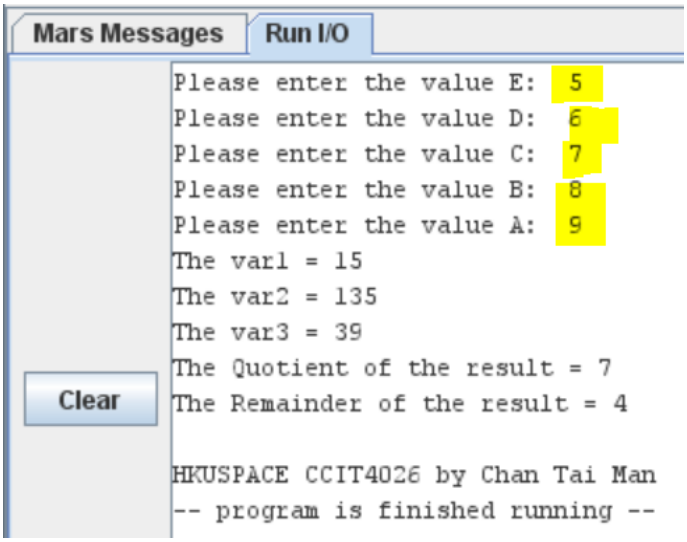
Step 9.4 # Store the remainder of the division result to the variable *remainder*.

Step 9.5 # Print all the intermediate results according to the sample output format.

Step 9.6 # Print the Author message.

Step 9.7 # Exit from the program.

Sample **Input** and Output



THE ABOVE FIVE INPUT VALUES ARE EXAMPLES ONLY.

USE OTHER VALUES WITHIN THE RANGE !!!!

Submission instructions:

i. Based on the template program in "Lab3_mon_template.asm", write the MIPS assembly program to complete the lab. tasks above.

ii. **Strictly follow** the steps shown in the Pseudo-code.

iii. Take each of the step descriptions shown in the **Pseudo code as comments** in your program.

iv. Complete each step with your MIPS instructions.

v. Copy and paste the screen capture of the source code from the MARS editor onto the provided answer sheet template in WORD file.

vi. Copy and paste the screen capture of the assembled window that shows your assembled machine codes.

vii. Copy and paste the screen capture of the "Run I/O" window that shows your program execution output (Program termination message MUST be shown).

viii. Fill in your name, class, student ID and a photo of your student ID card at the beginning of the WORD file. A sample submission format is shown below.

ix. Convert the provided answer sheet template WORD file to PDF format as your lab. report.

x. Rename your revised ".asm" and lab. report ".pdf" file to the file names according to the following format,

   i. *"ICO_Lab3_CL<classNO>_<yourname>_<studentID>.asm"and*
   ii. *"ICO_Lab3_CL<classNO>_<yourname>_<studentID>.pdf"*
   iii. e.g., For CL01,
      1. *"ICO_Lab3_CL01_CHANTaiMan_20002000.asm"and*
      2. *"ICO_Lab3_CL01_CHANTaiMan_20002000.pdf"*

xi. Upload BOTH the lab. report PDF file and your ".asm" to SOUL before the deadline.

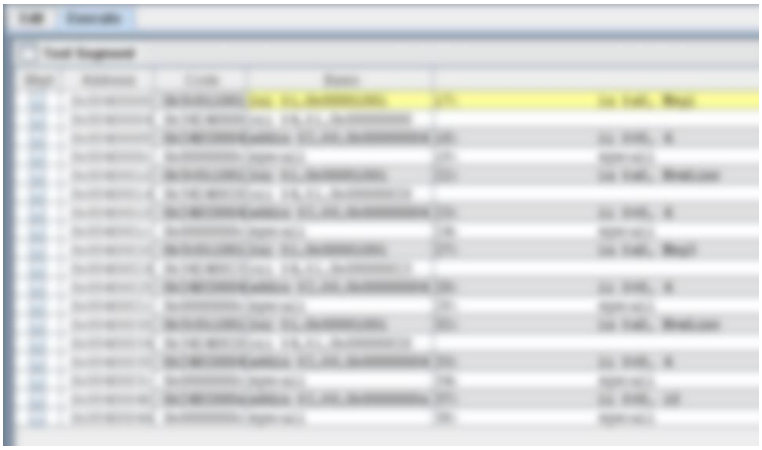
xii. No mark would be given if no source code file ".asm" is submitted OR missing your student ID card photo.

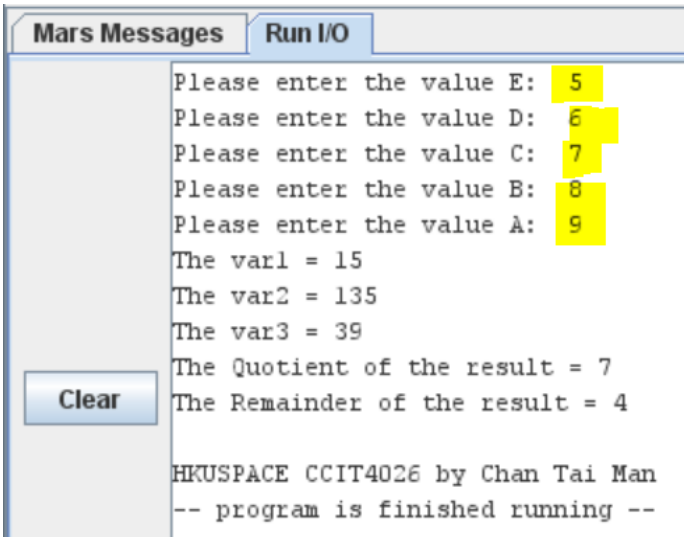
xiii. Late submission will NOT be graded.

Submission Sample:

		(Insert a photo of your student ID card below)
Student Name:		(No mark would be given if no student ID is present here)
Student no.:		
Class Number:		

Source Code (Captured from MARS)	Screen capture ALL the source code directly from MARS Editor.

	
Assembled Results (Captured from MARS)	Screen capture ALL the output directly from "Execute – Text Segment" window: 
The Formula	$$Result = (quotient{:}remainder) = \frac{((B + C) * A - (D \ll 4))}{E}$$
Manual calculations	Pick the five integers according to the following constraint, $$5 \leq E < D < C < B < A \leq 20$$
	$E = 5$ $D = 6$ $C = 7$ $B = 8$ $A = 9$ $var1 = (B + C) = 15$ $var2 = var1 * A = 135$

	$var3 = var2 - (A \ll 4) = 39$ $quotient = var3 \,/\, E = 7$ $remainder = var3 \,\%\, E = 4$ (All results are integers)
Run I/O results (Captured from MARS)	Screen capture the output directly from "Run I/O" window: 

— THE END —