CMPE 140 – Laboratory Assignment 3

Hyeran Jeon
Computer Engineering Department, San Jose State University
(This lab is created by Prof. Donald Hung)

MIPS Instruction Set Architecture & Programming (2)

Purpose

Gain familiarity with MIPS ISA control structures and the $hi and $lo registers.

Tasks

1) Write a MIPS assembly program to perform the arithmetic computation shown in the following C++ pseudo code. Note that the C++ pseudo code has several variables (*a*, *b*, *c*, *x*, and *y*). Use the following registers to store the values of these variables:

   $a0 ← a
   $a1 ← b
   $s0 ← c
   $s1 ← x
   $s2 ← y

   - Variables initialization

```
1.   a = 0x8000;      #MIPS instruction:      ori $a0, $0, 0x8000
2.   b = 0x00A9;
3.   c = 1974;
```

   - Arithmetic computation

```
4.   x = a * a;
5.   store the value of x to memory location at address 0x20;
6.   y = x * b;
7.   store the value of y to memory location at address 0x24;
8.   y = y >> 16;
9.   c = (c + y / c) / 2;
10.  store the value of c to memory location at address 0x2C;
```

   - While loop

```
11.  while(c >= 1665){
12.       c = (c + y / c) / 2;
13.  }
14.  c = c << 8;
15.  store the value of c to memory location at 0x30;
```

   Requirements: Use no more than 28 real MIPS instructions.

2) Assemble your MIPS assembly code and single-step execute through all instructions. After the execution of each instruction, verify the contents of the relevant registers. Record the execution results using the test log table on page 3 of the assignment, and note the value at the following memory addresses when the program execution has completed:

   - 0x20 – 0x23;
   - 0x24 – 0x27;

- 0x28 – 0x2b;
- 0x2c – 0x2f;
- 0x30 – 0x33;

3) Write a MIPS assembly program to calculate the factorial of a given integer *n*. The factorial of *n* is defined as

   $n! = n*(n-1)*\ldots*1$

   Note that 0! = 1.

   Algorithm for computing factorial:

```
1. INPUT n = 5; //given number n
2. f = 1;
3. while (n > 1)
   {
         f = f * n;
         n = n -1;
   }
4. OUTPUT f; //factorial f = n!
```

   Requirements:

   1. Input number *n* = 5, to be stored in memory location at address 0x00.
   2. Register assignment: `$a0` ← `n`; `$s0` ← `n!`
   3. You must use the algorithm shown above.
   4. The assembly program shall contain no more than 11 real MIPS instructions.
   5. The factorial of 5 must be written to the memory location at address 0x10.

4) Assemble the MIPS assembly code, single-step execute through each instruction and verify the contents of the relevant registers after each instruction's execution. Record the execution results using the test log table on page 4, and indicate the value at the following memory addresses when the entire program is executed:

   - 0x00 – 0x03;
   - 0x10 – 0x13;

5) Write your lab report. It should include the source code, the recorded test results (typed test logs), screen captures of the appropriate execution windows generated by the assembler, and a conclusion/discussion section.

CMPE140 Lab 3 Task 1 Test Log
Algorithm 1

Programmer's Name: _____

Checked by:_____, Date:_____

Adr	MIPS Instruction	Machine Code	Registers				
			$a0	$a1	$s0	$s1	$s2
00							
04							
08							
0c							
10							
14							
18							
1c							
20							
24							
28							
2c							
30							
34							
38							
3c							
40							
44							
48							
4c							
50							
54							
58							
5C							
60							
64							
68							
6C							

Memory contents			
Word @ 0x20	Word @ 0x24	Word @ 0x2C	Word @ 0x30

CMPE140 Lab 3 Task 2 Test Log
Algorithm 2

Programmer's Name: _____

Checked by:_____, Date:_____

Adr	MIPS Instruction	Machine Code	Registers				Memory Content	
			$a0	**$s0**	**$**	**$**	Word @ 0x00	Word @ 0x10
00								
04								
08								
0c								
10								
14								
18								
1c								
20								
24								
28								