Computer Architecture HW 02

(2012 fall)

4th Edition Exercise 2.4.1-3

The Following problems deal with translating from C to MIPS. Assume that the variables f, g, h, i, and j are assigned to registers $s0, $s1, $s2, $s3, and $s4, respectively. Assume that the base address of the arrays A and B are in registers $s6, and $s7, respectively.

```
a. f = g + h + B[4];
b. f = g – A[B[4]];
```

2.4.1. For the C statements above, what is the corresponding MIPS assembly code?

2.4.2. For the C statements above, how many MIPS assembly instructions are needed to perform the C statement?

2.4.3. For the C statements above, how many different registers are needed to carry out the C statement?

4th Edition Exercise 2.7.1-3

The following problems explore number conversions from signed and unsigned binary number to decimal numbers.

a. $1010\ 1101\ 0001\ 0000\ 0000\ 0000\ 0000\ 0010_{two}$

b. $1111\ 1111\ 1111\ 1111\ 1011\ 0011\ 0101\ 0011_{two}$

2.7.1. For the patterns above, what base 10 number does it represent, assuming that it is a two's complement integer?

2.7.2. For the patterns above, what base 10 number does it represent, assuming that it is an unsigned integer?

2.7.3. For the patterns above, what hexadecimal number does it represent?

4th Edition Exercise 2.11.1-3

In the following problems, the data table contains bits that represent the opcode of an instruction. You will be asked to translate the entries into assembly code and determine what format of MIPS instruction the bits represent.

a. 0XAE0BFFFC

b. 0X8D08FFC0

2.11.1. What binary number does the above hexadecimal number represent?

2.11.2. What decimal number does the above hexadecimal number represent?

2.11.3. What instruction does the above hexadecimal number represent?

4th Edition Exercise 2.16.1-3

For these problems, the table holds various binary values for register $t0. Given the value of $t0, you will be asked to evaluate the outcome of different branches.

a. $1010\ 1101\ 0001\ 0000\ 0000\ 0000\ 0000\ 0010_{two}$

b. $1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111_{two}$

2.16.1. Suppose that register $t0 contains a value from above and $t1 has the value

$$0011\ 1111\ 1111\ 1000\ 0000\ 0000\ 0000\ 0000_{two}$$

What is the value of $t2 after the following instructions?

```
        slt    $t2, $t0, $t1
        beq    $t2, $zero, ELSE
        J      DONE
ELSE: addi $t2, $zero, 2
DONE:
```

2.16.2. Suppose that register $t0 contains a value from the table above and is compared against the value X, if any, will $t2 be equal to 1? Specify in a range of values.

```
        slti $t2, $t0, X
```

2.16.3. Suppose the program counter (PC) is set to 0x0000 0020. Is it possible to use the jump (`j`) MIPS assembly instruction to set the PC to the address as shown in the data table above? Is it possible to use the branch-on-equal (`beq`) MIPS assembly instruction to set the PC to the address as shown in the data table above?

[Answer in the form of "`jump`-(yes/no), `beq`-(yes/no)"]