Exercise 2

2013-10-15 280 Points

Pls. mail your homework (converted into PDF File) to coa.2012.assignment@gmail.com.

File name: ID_YourName_HW_Sequence.pdf, ex: **5zzzzzzz123_Obama_HW_01.pdf**（You could save the doc/docx as PDF in word 2007 or later, using ‘Save as…’）

Exercise 2.4

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 [10] <2.2, 2.3> For the C statements above, what is the corresponding MIPS assembly code?

2.4.2 [5] <2.2, 2.3> For the C statements above, how many MIPS assembly instructions are needed to perform the C statement?

2.4.3 [5] <2.2, 2.3> For the C statements above, how many different registers are needed to carry out the C statement?

The following problems deal with translating from MIPS to C. 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.	`add $s0, $s0, $s1`
	`add $s0, $s0, $s2`
	`add $s0, $s0, $s3`
	`add $s0, $s0, $s4`

b.	`lw $s0, 4($s6)`

2.4.4 [10] <2.2, 2.3> For the MIPS assembly instructions above, what is the corresponding C statement?

2.4.5 [5] <2.2, 2.3> For the MIPS assembly instructions above, rewrite the assembly code to minimize the number of MIPS instructions (if possible) needed to carry out the same function.

2.4.6 [5] <2.2, 2.3> How many registers are needed to carry out the MIPS assembly as written above? If you could rewrite the code above, what is the minimal number of registers needed?

Exercise 2.10

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.	$1010\ 1110\ 0000\ 1011\ 0000\ 0000\ 0000\ 0100_{two}$
b.	$1000\ 1101\ 0000\ 1000\ 0000\ 0000\ 0100\ 0000_{two}$

2.10.1 [5] <2.5> For the binary entries above, what instruction do they represent?

2.10.2 [5] <2.5> What type (I-type, R-type) instruction do the binary entries above represent?

2.10.3 [5] <2.4, 2.5> If the binary entries above were data bits, what number would they represent in hexadecimal?

In the following problems, the data table contains MIPS instructions. You will be asked to translate the entries into the bits of the opcode and determine what is the MIPS instruction format.

a.	`add $t0, $t0, $zero`
b.	`lw $t1, 4($s3)`

2.10.4 [5] <2.4, 2.5> For the instructions above, show the hexadecimal representation of these instructions.

2.10.5 [5] <2.5> What type (I-type, R-type) instruction do the instructions above represent?

2.10.6 [5] <2.5> What is the hexadecimal representation of the opcode, rs, and rt fields in this instruction? For R-type instructions, what is the hexadecimal representation of the rd and funct fields? For I-type instructions, what is the hexadecimal representation of the immediate field?

Exercise 2.12

In the following problems, the data table contains various modifications that could be made to the MIPS instruction set architecture. You will investigate the impact of these changes on the instruction format of the MIPS architecture.

a.	8 registers
b.	10 bit immediate constants

2.12.1

[5 Points] If the instruction set of the MIPS processor is modified, the instruction format must also be changed. For each of the suggested changes above, show the size of the bit fields of an R-type format instruction. What is the total number of bits needed for each instruction?

2.12.2

[5 Points] If the instruction set of the MIPS processor is modified, the instruction format must also be changed. For each of the suggested changes above, show the size of the bit fields of an I-type format instruction. What is the total number of bits needed for each instruction?

2.12.3

[5 Points] Why could the suggested change in the table above decrease the size of a MIPS assembly program? Why could the suggested change in the table above increase the size of a MIPS assembly program?

In the following problems, the data table contains hexadecimal values. You will be asked to determine what MIPS instruction the value represents, and find the MIPS instruction format.

a.	0x00601020
b.	0x8C430014

2.12.4

[5 Points] For the entries above, what is the value of the number in decimal?

2.12.5

[5 Points] For the hexadecimal entries above, what instruction do they represent?

2.12.6

[5 Points] What type(I-type, R-type) instruction do the binary entries above represent? What is the value of the op field and the rt field?

Exercise 2.13

In the following problems, the data table contains the values for registers `$t0` and `$t1`. You will be asked to perform several MIPS logical operations on these registers.

a.	`$t0 = 0x55555555, $t1 = 0x12345678`
b.	`$t0 = 0xBEADFEED, $t1 = 0xDEADFADE`

2.13.1 [5] <2.6> For the lines above, what is the value of `$t2` for the following sequence of instructions:

```
sll $t2, $t0, 4
or $t2, $t2, $t1
```

2.13.2 [5] <2.6> For the values in the table above, what is the value of `$t2` for the following sequence of instructions:

```
sll $t2, $t0, 4
andi $t2, $t2, -1
```

2.13.3 [5] <2.6> For the lines above, what is the value of `$t2` for the following sequence of instructions:

```
srl $t2, $t0, 3
andi $t2, $t2, 0xFFEF
```

In the following exercise, the data table contains various MIPS logical operations. You will be asked to find the result of these operations given values for registers `$t0` and `$t1`.

a.	`sll $t2, $t0, 1` `or $t2, $t2, $t1`
b.	`srl $t2, $t0, 1` `andi $t2, $t2, 0x00F0`

2.13.4 [5] <2.6> Assume that `$t0` = 0x0000A5A5 and `$t1` = 00005A5A. What is the value of `$t2` after the two instructions in the table?

2.13.5 [5] <2.6> Assume that `$t0` = 0xA5A50000 and `$t1` = A5A50000. What is the value of `$t2` after the two instructions in the table?

2.13.6 [5] <2.6> Assume that `$t0` = 0xA5A5FFFF and `$t1` = A5A5FFFF. What is the value of `$t2` after the two instructions in the table?

Exercise 2.16

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 [5] <2.7> 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 [5] <2.7> Suppose that register $t0 contains a value from the table above and is compared against the value X, as used in the MIPS instruction below. For what values of X, if any, will $t2 be equal to 1?

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

2.16.3 [5] <2.7> 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?

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.	0x00001000
b.	0x20001400

2.16.4 [5] <2.7> Suppose that register $t0 contains a value from above. What is the value of $t2 after the following instructions?

```
      slt  $t2, $t0, $t0
      bne  $t2, $zero, ELSE
      j    DONE
ELSE: addi $t2, $t2, 2
DONE:
```

2.16.5 [5] <2.6, 2.7> Suppose that register `$t0` contains a value from above. What is the value of `$t2` after the following instructions?

```
sll $t0, $t0, 2
slt $t2, $t0, $zero
```

2.16.6 [5] <2.7> Suppose the program counter (PC) is set to 0x2000 0000. 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?

Exercise 2.18

For these problems, the table holds some C code. You will be asked to evaluate these C code statements in MIPS assembly code.

a.	`for(i=0; i<10; i++)` `    a += b;`

b.	`while (a < 10){`
	`    D[a] = b + a;`
	`    a += 1;`
	`}`

2.18.1 [5] <2.7> For the table above, draw a control-flow graph of the C code.

2.18.2 [5] <2.7> For the table above, translate the C code to MIPS assembly code. Use a minimum number of instructions. Assume that the value a, b, i, j are in registers $s0, $s1, $t0, $t1, respectively. Also, assume that register $s2 holds the base address of the array D.

2.18.3 [5] <2.7> How many MIPS instructions does it take to implement the C code? If the variables a and b are initialized to 10 and 1 and all elements of D are initially 0, what is the total number of MIPS instructions that is executed to complete the loop?

For these problems, the table holds MIPS assembly code fragments. You will be asked to evaluate each of the code fragments, familiarizing you with the different MIPS branch instructions.

a.

```
       addi  $t1, $0, 100
LOOP:  lw    $s1, 0($s0)
       add   $s2, $s2, $s1
       addi  $s0, $s0, 4
       subi  $t1, $t1, 1
       bne   $t1, $0, LOOP
```

b.

```
       addi  $t1, $s0, 400
LOOP:  lw    $s1, 0($s0)
       add   $s2, $s2, $s1
       lw    $s1, 4($s0)
       add   $s2, $s2, $s1
       addi  $s0, $s0, 8
       bne   $t1, $s0, LOOP
```

2.18.4 [5] <2.7> What is the total number of MIPS instructions executed?

2.18.5 [5] <2.7> Translate the loops above into C. Assume that the C-level integer i is held in register $t1, $s2 holds the C-level integer called result, and $s0 holds the base address of the integer MemArray.

2.18.6 [5] <2.7> Rewrite the loop in MIPS assembly to reduce the number of MIPS instructions executed.

Exercise 2.26

For this exercise, you will explore the range of branch and jump instructions in MIPS. For the following problems, use the hexadecimal data in the table below.

a.	0x00001000
b.	0xFFFC0000

2.26.1 [10] <2.6, 2.10> If the PC is at address 0x00000000, how many branch (no jump instructions) do you need to get to the address in the table above?

2.26.2 [10] <2.6, 2.10> If the PC is at address 0x00000000, how many jump instructions (no jump register instructions or branch instructions) are required to get to the target address in the table above?

2.26.3 [10] <2.6, 2.10> In order to reduce the size of MIPS programs, MIPS designers have decided to cut the immediate field of I-type instructions from 16 bits to 8 bits. If the PC is at address 0x0000000, how many branch instructions are needed to set the PC to the address in the table above?

For the following problems, you will be using making modifications to the MIPS instruction set architecture.

a.	8 registers
b.	10 bit immediate/address field

2.26.4 [10] <2.6, 2.10> If the instruction set of the MIPS processor is modified, the instruction format must also be changed. For each of the suggested changes above, what is the impact on the range of addresses in a beq instruction? Assume that all instructions remain 32 bits long and any changes made to the instruction format of I-type instructions only increase/decrease the immediate field of the beq instruction.

2.26.5 [10] <2.6, 2.10> If the instruction set of the MIPS processor is modified, the instruction format must also be changed. For each of the suggested changes above, what is the impact on the range of addresses a jump instruction? Assume that instructions remain 32 bits long and any changes made to the instruction format of J-type instructions only impact the address field of the jump instruction.

2.26.6 [10] <2.6, 2.10> If the instruction set of the MIPS processor is modified, the instruction format must also be changed. For each of the suggested changes above, what is the impact on the range of addresses a jump register instruction, assuming that each instruction must be 32 bits.

Exercise 2.39

The CPI of the different instruction types is given in the following table.

	Arithmetic	Load/Store	Branch
a.	2	10	3
b.	1	10	4

2.39.1 [5] <2.18> Assume the following instruction breakdown given for executing a given program:

	Instructions (in millions)
Arithmetic	500
Load/Store	300
Branch	100

What is the execution time for the processor if the operation frequency is 5 GHz?

2.39.2 [5] <2.18> Suppose that new, more powerful arithmetic instructions are added to the instruction set. On average, through the use of these more powerful arithmetic instructions, we can reduce the number of arithmetic instructions needed to execute a program by 25%, and the cost of increasing the clock cycle time by only 10%. Is this a good design choice? Why?

2.39.3 [5] <2.18> Suppose that we find a way to double the performance of arithmetic instructions? What is the overall speed-up of our machine? What if we find a way to improve the performance of arithmetic instructions by 10 times!?

The following table shows the proportions of instruction execution for the different instruction types.

	Arithmetic	Load/Store	Branch
a.	60%	20%	20%
b.	80%	15%	5%

2.39.4 [5] <2.18> Given the instruction mix above and the assumption that an arithmetic instruction requires 2 cycles, a load/store instruction takes 6 cycles, and a branch instruction takes 3 cycles, find the average CPI.

2.39.5 [5] <2.18> For a 25% improvement in performance, how many cycles, on average, may an arithmetic instruction take if load/store and branch instructions are not improved at all?

2.39.6 [5] <2.18> For a 50% improvement in performance, how many cycles, on average, may an arithmetic instruction take if load/store and branch instructions are not improved at all?