NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

EXAMINATION FOR

Semester 2 AY2010/2011

CS2100 – COMPUTER ORGANISATION

April 2011 **Time allowed: 2 hours**

INSTRUCTIONS TO CANDIDATES

1. This examination paper consists of **SIXTEEN (16)** questions and comprises **NINE (9)** printed pages.
2. This is an **OPEN BOOK** examination.
3. Answer all questions.
4. Write your answers in the **ANSWER BOOK** provided.
5. Fill in your Matriculation Number with a pen, clearly on odd-numbered pages of your ANSWER BOOK.
6. You may use pencil to write your answers.
7. You are to submit only the ANSWER BOOK and no other document.

Questions 1 - 12: Each question has only one correct answer. Write your answers in the boxes provided in the Answer Book. One mark is awarded for a correct answer and no penalty for wrong answer.

1. Given the following hexadecimal form of a value in IEEE 754 floating-point format:

   45900000

   what decimal value does it represent?

   A. 4608
   B. 4590
   C. 2304
   D. 512
   E. None of the above

2. The table below shows the number of cycles for each instruction class and their frequencies in a program.

Instruction class	A	B	C	D
CPI	2	3	?	?
Frequency	40%	30%	20%	10%

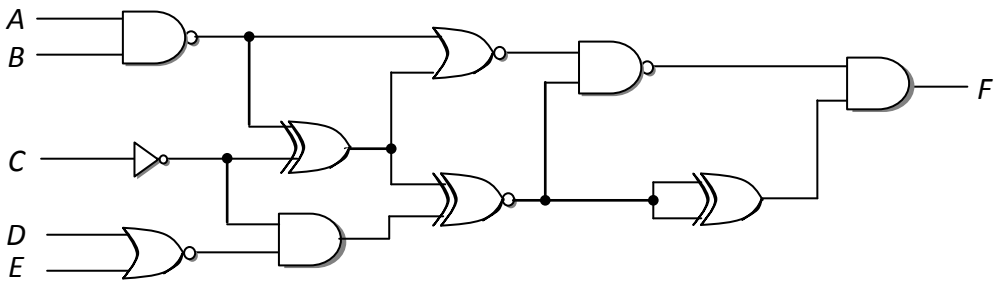
   If the average CPI is **2.7**, what are the CPIs of instruction classes C and D?

   A. C = 2; D = 3
   B. C = 2; D = 4
   C. C = 3; D = 3
   D. C = 3; D = 4
   E. C = 4; D = 4

3. Following from question 2, if the machine the program runs on has a clock frequency of 500MHz, and the program contains 1 million instructions, how long does the program run in this machine?

   A. 1.35 milliseconds
   B. 5.4 milliseconds
   C. 54 milliseconds
   D. 185 milliseconds
   E. 1.35 seconds

4. Given the following circuit, what is the value of ***F***?

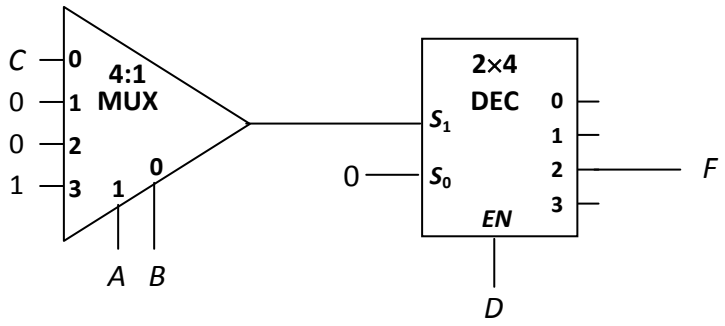


A. 0
B. 1
C. $B' \cdot C \cdot E + A' \cdot B \cdot D$
D. $A \cdot B' + (E' + A \cdot D') \cdot C'$
E. None of the above

5. The 5 stages of each instruction in an ISA take 2ns, 3ns, 2ns, 3ns and 1ns to execute. What is the **speedup** of a 5-stage pipelined system compared to a non-pipelined single-cycle system in executing a code with 50 instructions?

A. 2.9
B. 3.4
C. 3.7
D. 5.0
E. 9.2

6. Given the following circuit, what is ***F(A,B,C,D)***?



A. 0
B. $m3$
C. $\Sigma m(3, 13, 15)$
D. $\Pi M(2, 3, 12, 13, 14, 15)$
E. None of the above

7. Given the PCSPIM code below to print the value 12 on the console:

```
# missing instruction here
li $v0, 1
syscall
```

Which of the following instructions could fill in the missing instruction?

A. **addi $v0, $zero, 12**
B. **addi $s0, $zero, 12**
C. **addi $t0, $zero, 12**
D. **addi $at, $zero, 12**
E. **addi $a0, $zero, 12**

8. The code below belongs to a stack architecture. @*n* refers to a memory location associated with variable *n*.

```
push  @a
push  @b
push  @c
add
push  @d
mul
add
pop   @e
```

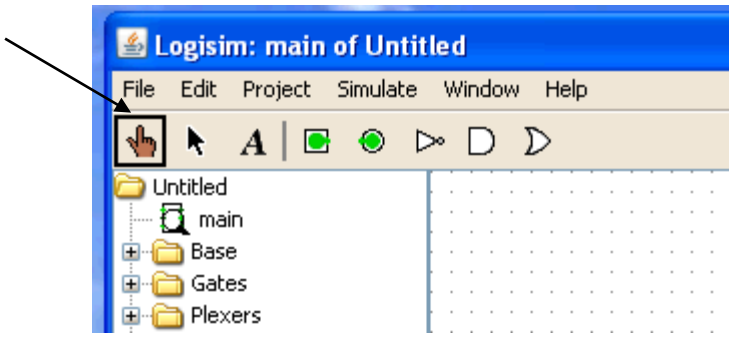
Which of the following high-level statements is likely to correspond to the above code?

A. **e = a * b + c + d;**
B. **e = a + b + c * d;**
C. **e = a + b * (c + d);**
D. **e = a + d * (b + c);**
E. None of the above

9. If you want to swap the lower 16 bits of a register's content with the upper 16 bits, for example, 0x12345678 becomes 0x56781234, which of the following MIPS instruction would be useful?

A. sll
B. rol
C. swap
D. lui
E. and

10. What is the name of the tool (the hand) indicated below?



A. HAND (Have A Nice Day)
B. Execute
C. Touch
D. Select
E. Poke

11. Is the following statement <u>true or false</u>?

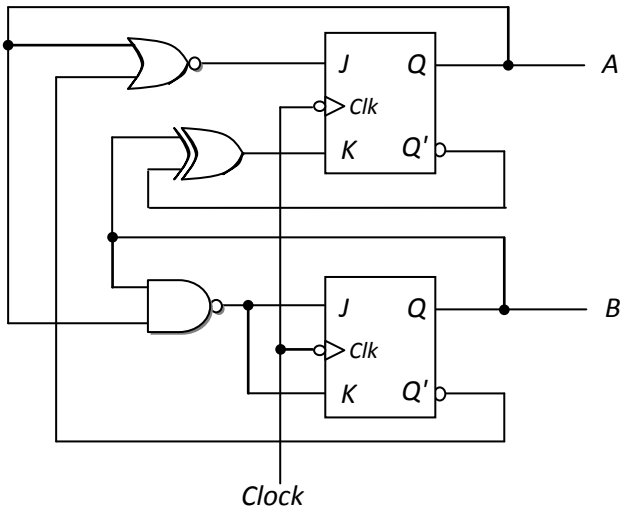
For a Boolean function *F(A,B,C,D)*, the expression ***F(A,B,C,D) = A·B'·C + A'·B·C'*** is a sum-of-minterms expression.

12. Is the following statement <u>true or false</u>?

The register **$t0** contains the value zero after the MIPS instruction below is executed:

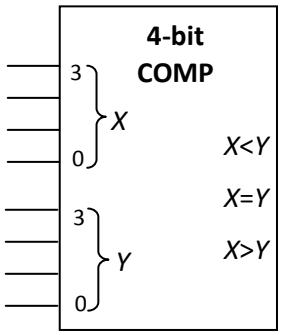
```
lw $t0, 0($zero)
```

13. The sequential circuit with state ***AB*** shown below contains two JK flip-flops. Complete the state diagram. [4 marks]



14. The figure below shows the block diagram of a **4-bit magnitude comparator**. Show how you could use a single 4-bit magnitude comparator and the fewest number of logic gates to compare two 4-bit 2s-complement numbers A (= $a_3a_2a_1a_0$) and B (= $b_3b_2b_1b_0$), and generate output F such that $F(A,B)$ = 1 if the actual value represented by A is smaller than the actual value represented by B, or 0 otherwise.

    For example, if A = 1010 (which represents -6) and B = 0011 (which represents 3), then output F should be 1, as $A < B$. [6 marks]



15. **[20 marks]**

The following MIPS code reads an integer array ***A***, whose base address is stored in **$s0**, and computes some answer in **$s2**. The register **$s1** is associated with the integer variable **cutoff**.

```
        addi $s2, $zero, 0     # Inst1
        addi $t0, $s0, 40      # Inst2
Here:   addi $t0, $t0, -4      # Inst3
        lw   $t1, 0($t0)       # Inst4
        slt  $t2, $t1, $s1     # Inst5
        beq  $t2, $zero, Skip  # Inst6
        addi $s2, $s2, 1       # Inst7
Skip:   beq  $t0, $s0, Fin     # Inst8
        j    Here              # Inst9
Fin:
```

a. How many elements of array ***A*** are accessed? [1 mark]

b. For the instruction **addi $t0, $t0, -4**, write out its hexadecimal representation. [1 mark]

c. For the only R-format instruction in the code, write out its hexadecimal representation. [2 marks]

d. Write an equivalent C code for this MIPS code. You may assume that the integer array name is ***A***, the integer variable **cutoff** is associated with **$s1**, and the integer variable **count** is associated with **$s2**. The array elements should be accessed in the same order as in the MIPS code. You may also assume that all variables have been declared. [3 marks]

e. Assuming that the first instruction **addi $s2, $zero, 0** is at address **0x00000040**, what is the immediate value, in decimal, in the **j Here** instruction? [1 mark]

f. Assuming that the first instruction **addi $s2, $zero, 0** is at address **0x8FFFFFF0**, explain why the **j Here** instruction will not work. Replace the **j Here** instruction with the **jr** instruction and other appropriate instructions. [3 marks]

The **jr** instruction has this format:

jr *rs*

where register *rs* contains the address of the instruction to jump to.

15. **(continued)**

The MIPS code is reproduced here.

The following MIPS code reads an integer array ***A***, whose base address is stored in **$s0**, and computes some answer in **$s2**. The register **$s1** is associated with the integer variable **cutoff**.

```
        addi $s2, $zero, 0     # Inst1
        addi $t0, $s0, 40      # Inst2
Here:   addi $t0, $t0, -4      # Inst3
        lw   $t1, 0($t0)       # Inst4
        slt  $t2, $t1, $s1     # Inst5
        beq  $t2, $zero, Skip  # Inst6
        addi $s2, $s2, 1       # Inst7
Skip:   beq  $t0, $s0, Fin     # Inst8
        j    Here              # Inst9
Fin:
```

g. The datapath discussed in class does not include the **jr** instruction. The inclusion of the **jr** instruction in the datapath would require a new control **JumpReg**, which is set to 1 for **jr** instruction, or 0 otherwise.

The **jr** is an R-format instruction. How is the **JumpReg** control used to ensure that the other R-format instructions (such as **add**, **slt**) work properly? [2 marks]

h. Assuming a MIPS pipeline with forwarding and branch is resolved in the ID stage with no branch prediction. Using the grid provided in the Answer Book, indicate the pipeline stage each instruction is in for each cycle, for the first iteration of the loop.

You may write 'F', 'D', 'E', 'M', 'W' for the Instruction Fetch, Instruction Decode, Execute, Memory, and Writeback stages respectively. For stall, you may write 'S' or '– ', and explain how the stall comes about. [3 marks]

i. If we implement branch prediction, which of these two choices would you choose: **branch taken** or **branch not taken**, and why? Based on your choice, how would the values in the array affect the performance? [2 marks]

j. To reduce the number of stalls in (h), two of the instructions in the given MIPS code can be swapped, with appropriate modification to their operands. Indicate which two old instructions can be swapped, and write down the two new instructions. Explain how the new code reduces the number of stalls. [2 marks]

16. **[8 marks]**
The following MIPS code is same as the one in question 15.

```
        addi $s2, $zero, 0     # Inst1
        addi $t0, $s0, 40      # Inst2
Here:   addi $t0, $t0, -4      # Inst3
        lw   $t1, 0($t0)       # Inst4
        slt  $t2, $t1, $s1     # Inst5
        beq  $t2, $zero, Skip  # Inst6
        addi $s2, $s2, 1       # Inst7
Skip:   beq  $t0, $s0, Fin     # Inst8
        j    Here              # Inst9
Fin:
```

Assuming that the above code is executed in a MIPS machine with a **direct-mapped instruction cache** with 4 blocks and a block size of 2 words.

a. Write out the width (number of bits) of the **index** field and **offset** field. [2 marks]

b. The cache is initially empty. Assuming that the first instruction **addi $s2, $zero, 0** is at address **0x00000040**, fill in the final content of the cache after the execution of the MIPS code.

You may write "Inst*X*", where *X*=1,2,...,9 to indicate instruction *X*. You may write the tag value in decimal. [4 marks]

c. How many cache misses are there in the execution of the MIPS code? What are the instructions that cause the misses? [2 marks]

=== END OF PAPER ===