ACS Applied Computer Science

THE UNIVERSITY OF WINNIPEG

APPLIED COMPUTER SCIENCE

ACS-2906-001
Computer Architecture and System Software

Fall 2024

Assignment 1

Due date: October 13, 2024 11:59 pm **Total marks: 100**

Questions

1) Registers

Motivation: The focus of this question is to help you become familiar with the CPU and its registers, all of which applies directly when writing assembly language programs.

Question: List and explain the registers for a 32-bit Intel Architecture (IA) processor. Specifically, your answer should include a description of the data registers (also called general purpose registers), pointer and index registers, and the control registers. Also, give a description of the execution cycle and the system clock. Cite any sources you used to answer this question.

Hint: Review chapters 2 and 3 of the (UofW electronic resource) S. Dandamudi, *Introduction to Assembly Language Programming for Pentium and RISC Processors*, Springer, 2005. You can access this resource using this link.

Evaluation: This question is worth **12 marks**. 8 marks for the register discussion, half of each mark is for correctly identifying a register, and the other half will be awarded based on whether your description matches the actual function of the register. Your answers for the description of the execution cycle and system clock are each worth 2 marks.

2) Boolean operations

Motivation: Bit-level representations and manipulations is a skill needed by all programmers and is not just for assembly language programming. Consequently, the focus of this question is to help you gain experience with bit-level, logic, and shift operators.

Question: Perform the following bit-level operations on 8-bit unsigned integers (all integers below appear in base 10):

a) `121 & 217`
b) `~245`
c) `~( 51 | 13 )`
d) `~ ( 16 & (8 | 15 ) )`
e) `~(61 >> 3)` (Logical shift)
f) `( ( 46 | 22) << 4 )`

Evaluation: This question is worth **12** marks (2 marks per subproblem). You **MUST** show your work to receive full marks.

3) Binary Numbers

Problem: Perform the following operations in binary using 8 bits for the integer part and 5 bits for fractional part.

1. $-80 + 42$
2. $-99 - 20$
3. $60 - 70$
4. $-59 \div 3$
5. 52×5
6. $(0.0101)_2 \times (0.11)_2$
7. $(0.0101)_2 \div (0.001)_2$

Evaluation: This question is worth **10** marks. The addition and subtraction questions are 1 marks each and the multiplication and division questions are 2 marks each. The other two problems are 1.5 marks each. You **MUST** show your work to receive marks.

4) Floating point

Motivation: The focus of this question is to improve the understanding of the floating-point bit representation.

Convert the following numbers from decimal to binary floating point representation. Show all your calculations of *exp, s, frac, M, E* and *bias*. Use $k = 7, n = 8$ for your calculations.

a) $(2304)_{10}$

b) $(-1751)_{10}$

b) Perform the following operation in floating point 32-bit binary representation:

- $(-1751)_{10} - (2304)_{10}$

Evaluation: The conversions in part **a** are worth **6 marks**, and the operation in part **b** is worth **10 marks**. You **MUST** show your work to receive marks.

5) Combinatorics on Words

Background: Recall from the lecture notes that an integer x can be represented as a vector $[x_{w-1}, x_{w-2}, \ldots, x_0]$, where w is the number of bits used to represent x. As a fun exercise, let us consider a problem where we want to partition x into two new numbers, labelled y and z. We can do this by scanning through the bits of x from the least significant bit (lsb) to the most significant bit (msb). We form the new integers, y & z, according to the following rule:

- Anytime a 1 is encountered during the scan through x we place a 1 in either y or z in an alternating pattern
- Moreover, the bits are placed into the new integers at the same bit position as found in the original integer x.

. Below is an example.

Let $x = (13)_{10} = (1101)_2$. Then, scanning from lsb to msb gives the following result.

- Position 0: $x_0 = 1$. Thus, $y_0 = 1$, giving $y = (0001)_2$.
- Position 1: $x_1 = 0$. Thus, nothing happens.
- Position 2: $x_2 = 1$. Thus, $z_2 = 1$, giving $z = (0100)_2$.
- Position 3: $x_3 = 1$. Thus, $y_3 = 1$, giving $y = (1001)_2$.

The final result is that $y = (1001)_2 = (9)_{10}$ and $z = (0100)_2 = (4)_{10}$.

Problem: Write a Java program that takes an integer x as an input and produces y and z as an output.

Sample Execution:

```
Enter an integer x
13
y z
9 4
```

Hint: In Java, you can iterate over the bits of a number using:

```
for (int i = 0; (x >>> i) > 0; i++) {
    currentBit = (x >>> i)
}
```

Evaluation: This question is worth 50 marks, where 30 of the marks are for the algorithm, 5 are for adding input functionality to your program, and 5 are for adding output functionality. The remaining 10 marks are style points based on subjective evaluation of the quality of your solutions and your comments.

Hand in instructions

Include your name and student number in all files. Comment, comment, comment! We will deduct up to 5 marks for students that do not follow instructions or submit poorly formated work. Zip all the files into a single archive named `YourStudentID_Ass1.zip`. Submit the zip file using Nexus.