**COMP1411 (Spring 2023) Introduction to Computer Systems**

**REFEREENCE ANSWERS FOR ASSIGNMENT 1**

Individual Assignment 1 Duration: 00:00, 11-Feb-2023 ~ 23:59, 12-Feb-2023

**Question 1**. [2 marks]

Consider a 32-bit floating-point representation based on the IEEE floating-point format:

* the highest bit is used for the sign bit,
* the sign bit is followed by 6 exponent bits, which are then
* followed by 25 fraction bits.

1. **Convert** decimal value -28.40625 into the above 32-bit IEEE floating-point format. Write out the result in the hex-decimal form.

*Answer*:

STEP 1: the sign bit is 1

STEP 2: 28.40625 = 11100.01101 = 1.110001101 \* 24

STEP 3: the exp = 4 + 31 (bias) = 35 = 1000112

STEP 4: the frac part is 1100 0110 1000 0000 0000 0000 0 (25 binary bits)

STEP 5: the 32-bit floating-point number is 1100 0111 1000 1101 0000 0000 0000 0000, which written into the hex-decimal form is 0xC78D0000

1. Assume this 32-bit number is stored on a little-endian machine in the addresses 0x300~0x303. Please fill in the following table to show the byte stored in each address. To write a byte, please use the hex-decimal format starting with 0x.

|  |  |
| --- | --- |
| Address | Byte in the Address |
| 0x0300 | 0x00 |
| 0x0301 | 0x00 |
| 0x0302 | 0x8D |
| 0x0303 | 0xC7 |

**Question 2**. [0.5 marks]

Suppose that x and y are unsigned integers.

1. **Re-write** the following C-language statement only using << and – operations. Introducing new variables (other than x and y) is not allowed. Please show your steps.

**y = x \* 84;**

*Answer*:

STEP 1: 84 = 128 – 44 = 27 – 25 – 23 - 22

STEP 2: y = x \* 84 = x \* (27 – 25 – 23 - 22)

STEP 3: y = (x << 7) – (x << 5) – (x << 3) – (x << 2)

1. Given the C-language statement y = x \* b, assume that y, x, and b are all unsigned integers, and the result of x \* b does not overflow.

Can the statement y = x \* b always be re-written into a C-language statement only using << and – operations? Please answer yes or no, and explain why. Only answering yes or no without any explanation will receive a zero mark for this question.

*Answer*:

The answer is yes.

Assume that b can be encoded with N binary bits, then, we can always produce a number a, where a = 2N+1 – b. So, y = x \* b can always be re-written into “y = x \* (2N+1 – b)”. One step further, y = x \*b can always be re-written into a statement with only << and – operators.

**Question 3**. [1.5 marks]

Consider a 12-bit floating-point representation based on the IEEE floating-point format:

* the highest bit is used for the sign bit,
* the sign bit is followed by 4 exponent bits, which are then
* followed by 7 fraction bits.

1. What is the **largest positive normalized number** with the above floating-point format? Write the numbers in decimal form.
2. **Compute** the decimal value of the bit vector **0xD54** with the above floating-point format. Write the result in decimal form.

*Answer*:

1. The largest positive normalized number is: 0 1110 1111111

The value is 255.

1. STEP 1: 0xD54 = 1 1010 10101002

STEP 2: the sign bit: 1

STEP 3: the exp = 10102 = 10, Bias = 7, so E = 10 – 7 = 3

STEP 4: M = 1.10101

STEP 5: value = -13.25

NOTICE: There is an error in the second question in the final exam paper distributed, so all students who submits their assignment 1 all receives full mark for question 3(2).

**Question 4**. [1 mark]

Suppose that x, y, z, and a are all 16-bit unsigned integers.

1. Assume that the left-most bit is the highest bit. Write a single C-language statement to set the value of a such that:
   1. the left-most 5 bits of a are the same as the right-most 5 bits of x;
   2. the right-most 4 bits of a are the same as the left-most 4 bits of y;
   3. the middle 7 bits of z are the same as the left-most 7 bits of z.

Note that:

* You are only allowed to use logical bit shift operations and bit operations, including |, ^, and &, to set the value of a;
* NO arithmetic or if-then-else test (in any form) is allowed;
* Introducing new variables (other than x, y, z, and a) is NOT allowed;
* Using mask numbers is NOT allowed.

*Answer*:

a = (x << 11) | (y >> 12) | ((z >> 9) << 4)

1. If x = 0xDC9E, y = 0x36A7, and z = 0x9928, what the be the resulting value of a? Please write the value of a in hex-decimal form starting with the prefix 0x.

*Answer*:

x = 1101 1100 1001 1110

y = 0011 0110 1010 0111

z = 1001 1001 0010 1000

a = 1111 0100 1100 0011

a = 0xF4C3