**COMP1411 (Spring 2024) Introduction to Computer Systems**

**REFEREENCE ANSWERS FOR ASSIGNMENT 1**

Individual Assignment 1 Duration: 00:00, 24-Feb-2024 ~ 23:59, 25-Feb-2024

**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 5 exponent bits, which are then
* followed by 26 fraction bits.

1. **Convert** decimal value -29.21875 into the above 32-bit IEEE floating-point format. Write out the result in the hex-decimal format.

*Answer*:

STEP 1: the sign bit is 1

STEP 2: 29.21875 = 11101.00111 = 1.110100111 \* 24

STEP 3: the exp = 4 + 15 (bias) = 19 = 100112

STEP 4: the frac part is 1101 0011 1000 0000 0000 0000 00 (26 binary bits)

STEP 5: the 32-bit floating-point number is 1100 1111 0100 1110 0000 0000 0000 0000, which written into the hex-decimal form is 0xCF4E0000

1. Assume this 32-bit number is stored on a **big-endian** machine in the addresses 0x100~0x103. 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.

|  |  |
| --- | --- |
| Memory Address | Byte in the Address |
| 0x0100 | 0xCF |
| 0x0101 | 0x4E |
| 0x0102 | 0x00 |
| 0x0103 | 0x00 |

**Question 2**. [0.6 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 \* 78;**

*Answer*:

STEP 1: 78 = 128 – 50 = 27 – (25 + 24 + 21) = 27 – 25 - 24 - 21

STEP 2: y = x \* 78 = x \* (27 – 25 - 24 - 21)

STEP 3: y = (x << 7) – (x << 5) – (x << 4) – (x << 1)

**Question 3**. [1.4 marks]

Consider a 16-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 11 fraction bits.

1. What is the **largest positive normalized number** with the above floating-point format? Write the number in binary form.
2. **Compute** the decimal value of the bit vector 0x6D80 with the above floating-point format. Write the result in decimal format.

*Answer*:

1. The largest positive normalized number is: 0 1110 11111111111
2. STEP 1: 0x6D80 = 0110 1101 1000 00002

STEP 2: the sign bit: 0

STEP 3: the exp = 11012 = 13, Bias = 7, so E = 13 – 7 = 6

STEP 4: M = 1.10112

STEP 5: value = 1.10112 \* 26 = 108

**Question 4**. [1 mark]

Given the following C program:

图形用户界面, 文本

中度可信度描述已自动生成

1. What is the output of this program?
2. Explain why the output is generated in detail.

*Answer*:

1. The output is : b = -100
2. :

STEP 1: a = 0x9C = 100111002

STEP 2: the assignment “b = a” will let b exactly receive a’s bits, therefore, b = 10011100

STEP 3: the printf function takes “b” as input, and %d indicates that “b” will be explained with B2T or as 8-bit signed integer, then the output is: b = -100