Jalen Powell

1/28/2022

Comp 3350

COMP 3350 Project #1

Possible points: 100

Due: September 3, 2021 (Friday) **11:59pm CST (Central Standard Time)**

Goals:

- Get you familiar with data representation and simple logic operations for this course.

Requirements:

- Finish the questions section below. Points for each question included in parenthesis.
- Show your work to get full credit. **ZERO** point without steps for a result.
- Please start early. ZERO point for late submission. After the **11:59pm** on the due day, you can't submit your assignment anymore.
- Check deliverables section below. ZERO point for hand-written or scanned homework.

Deliverables:

- Save your solutions of questions as a **pdf** document. You can use this document as worksheet.
- Name document as a "**Firstname_Lastname.pdf**".
- Submit your "**Firstname_Lastname.pdf**" through the Canvas system. You do not need to submit hard copies.

Rebuttal period:

- You will be given a period of 2 **business** days to read and respond to the comments and grades of your homework or project assignment. The TA may use this opportunity to address any concern and question you have. The TA also may ask for additional information from you regarding your homework or project.

Questions:

1. (9 points) Convert the following unsigned base 2 numbers (binary) to base 16 numbers (hexadecimal):

A. <u>0110 0001 1111</u>

- 0110 = (0x2^0 + 1x2^1 + 1x2^2 + 0x2^3) → (6)_D → (6)_H
- 0001 = (1x2^0 + 0x2^1 + 0x2^2 + 0x2^3) → (1)_D → (1)_H
- 1111 = (1x2^0 + 1x2^1 + 1x2^2 + 1x2^3) → (15)_D → (F)_H
- => **(61F)__H**

<u>B. 1000 1111 1100</u>

- 1000 = (0x2^0 + 1x2^1 + 1x2^2 + 1x2^3) → (8)_D → (8)_H
- 1111 = (1x2^0 + 1x2^1 + 1x2^2 + 1x2^3) → (15)_D → (F)_H
- 1100 = (0x2^0 + 0x2^1 + 1x2^2 + 1x2^3) → (12)_D → (C)_H
- => **(8FC)_H**

<u>C. 0001 0110 0100 0101</u>

- 0001 = (1x2^0 + 0x2^1 + 0x2^2 + 0x2^3) → (1)_D → (1)_H
- 0110 = (0x2^0 + 1x2^1 + 1x2^2 + 0x2^3) → (6)_D → (6)_H
- 0100 = (0x2^0 + 0x2^1 + 1x2^2 + 0x2^3) → (4)_D → (4)_H
- 0101 = (1x2^0 + 0x2^1 + 0x2^2 + 0x2^3) → (5)_D → (5)_H
- => **(1645)_H**

2. (27 points)

(2.1) Convert the following binary

<u>a. 1100 1010</u>

- 0x2^0 + 1x2^1 + 0x2^2 + 1x2^3 + 0x2^4 + 0x2^5 + 1x2^6
- 2+8+64
- => **(-74)_D**

<u>b. 1111 0010</u>

- 0x2^0 + 1x2^1 + 0x2^2 + 0x2^3 + 1x2^4 + 1x2^5 + 1x2^6
- 2+16+32+64
- => **(-114)_D**

<u>c. 1000 0111</u>

- 1x2^0 + 1x2^1 + 1x2^2 + 0x2^3 + 0x2^4 + 0x2^5 + 0x2^6
- 1+2+4
- => **(-7)_D**

numbers into base 10 numbers (decimal), binary numbers are represented in signed magnitude representation.

(2.2) Redo the question 2.1, if the binary number are represented in One's complement representation.

<u>A) 1100 1010</u>

- (1011 0101) – 1's complement
- -(1x2^0 + 0x2^1 + 1x2^2 + 0x2^3 + 1x2^4 + 1x2^5 + 1x2^6)
- => **(-53)_D**

<u>B) 1111 0010</u>

- (1000 1101) – 1's complement

- -(1x2^0 + 0x2^1 + 1x2^2 + 1x2^3)
- => **(-13)_D**

<u>C) 1000 0111</u>

- (1111 1000) – 1's complement
- -(1x2^3 + 1x2^4 + 1x2^5 + 1x2^6)
- => **(-120)_D**

(2.3) Redo the question 2.1, if the binary number are represented in Two's complement representation.

<u>A) 1100 1010</u>

- (1011 0110) – 2's complement
- -(1x2^1 + 1x2^2 + 1x2^4 + 1x2^5)
- => **(-54)_D**

<u>B) 1111 0010</u>

- (1000 1110) – 2's complement
- -(1x2^1 + 1x2^2 + 1x2^3)
- => **(-14)_D**

<u>C) 1000 0111</u>

- (1111 1001) – 2’s complement
- -(1x2^0 + 1x2^3 + 1x2^4 + 1x2^5 + 1x2^6)
- => **(-121)_D**

For example, question (2.1), if 1100 1010 is a binary number represented in signed magnitude representation, what is the decimal value? Also do it again if 1100 1010 is a binary number in one’s complement representation and two’s complement representation. There 9 questions in total.

3. (36 points, answer 12 questions in total.)

(3.1) Convert the following base 10 (decimal) values to binary numbers (8-bits):

a. **-100_d**

- 2 – 100 → 0
- 2 – 50 → 0
- 2 – 25 → 1
- 2 – 12 → 0
- 2 – 6 → 0
- 2 – 3 → 1
- 2 – 1 → 1
- => **11100100**

b. **-16_d**

- 2 – 16 → 0
- 2 – 8 → 0
- 2 – 4 → 0
- 2 – 2 → 0
- 2 – 1 → 1
- => **10010000**

c. **-21_d**

- 2 – 21 → 1
- 2 – 10→ 0
- 2 – 5 → 1
- 2 – 2 → 0
- 2 – 1 → 1
- => **10010101**

d. **-0_d**

- => **10000000**

Each binary result represented in Signed magnitude representation.

(3.2) Redo the question (3.1), convert binary into in One's complement representation.

A) -100d

- 11100100
- => **10011011 – 1's complement**

B) -16d

- 10010000
- => **11101111 – 1's complement**

C) -21d

- 10010101
- => **11101010 – 1's complement**

D) -0d

- 10000000
- => **11111111 – 1's complement**

(3.3) Redo the question (3.1), convert binary into in Two's complement representation.

A) -100d

- 11100100
- => **10011100 – 2's complement**

B) -16d

- 10010000
- => **11110000 – 2's complement**

C) -21d

- 10010101
- => **11101011 – 2's complement**

D) -0d

- 10000000
- => **10000000 – 2's complement**

4. (4 points) What is the range of:

A. An unsigned 7-bit number?

- 0 – (2^n -1), n=7
- 0 – (2^7 -1)
- 0 – (128 -1)
- => **0 to 127**

B. A signed 7-bit number?

- -2^(n-1) – (2^(n-1) -1)
- -2^(7-1) – (2^(7-1) -1)
- -2^6 – (2^6 – 1)
- **=> -64 to 63**

5. (12 points) Provide the answer to the following problems (∧ = AND, ∨ = OR)

1. 1000 ∧ 1110
    - **=>1000 ∧ 1110 = 1000**
2. 1000∨1110
    - **=> 1000 ∨ 1110 = 1110**
3. (1000∧1110) ∨ (1001∧1110)
    - 1000 ∧ 1110 = 1000
    - 1001 ∧ 1110 = 1000
    - **=>1000 ∨ 1000 = 1000**

6. (9 points) Please demonstrate each step in the calculation of the arithmetic operation 25 - 65. (both 25 and 65 are signed decimal numbers)

- 25
- 2 – 25 → 1
- 2 – 12 → 0
- 2 – 6 → 0
- 2 – 3 → 1
- 00011001

- -65
- 2 – 65
- 2 – 32
- 2 – 16
- 2 – 8
- 2 – 4
- 2 – 2
- 1
- 11000001
- (111000001) – 2's complement

- 00011001 – 2’s complement

(25) 00011001

(-65) + 10111111

=> **(10101000) – 2’s complement → (-40)_D**

7. (3 points) Mathematically the answer in Q6 is -40_d. Please verify your answer in Q6 using a conversion of 2’s and decimal numbers.

- (-40)_D → (10101000) - 2’s complement

- -(0x2^0 + 0x2^1 + 0x2^2 + 1x2^3 + 0x2^4 + 1x2^5 + 0x2^6)
- -(8 + 32)
- => **(-40)_D**