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. 0110 0001 1111

0110 = 6, 0001 = 1, 1111 = F

= 61F (base 16)

B. 1000 1111 1100

1000 = 8, 1111 = F, 1100 = C

= 8FC (base 16)

C. 0001 0110 0100 0101

0001 = 1, 0110 = 6, 0100 = 4, 0101 = 5

= 1645 (base 16)

2. (27 points)

(2.1) Convert the following binary

a. 1100 1010

b. 1111 0010

c. 1000 0111

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

(a) 1 = negative -> 100 1010 = magnitude

$(1x2^6) + (0x2^5) + (0x2^4) + (1x2^3) + (0x2^2) + (1x2^1) + (0x2^0)$

= (64 + 8 + 2)

= 74 -> magnitude -> add sign

= -74 (base 10)

(b) 1 = negative -> 111 0010 = magnitude

$(1x2^6) + (1x2^5) + (1x2^4) + (0x2^3) + (0x2^2) + (1x2^1) + (0x2^0)$

= (64 + 32 + 16 +2)

= 114 -> magnitude -> add sign

= -114 (base 10)

(c) 1 = negative -> 000 0111 = magnitude

$(0x2^6) + (0x2^5) + (0x2^4) + (0x2^3) + (1x2^2) + (1x2^1) + (1x2^0)$

= (4 + 2 + 1)

= 7 -> magnitude -> add sign

= -7 (base 10)

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

(a) 1100 1010 -> keep msb -> invert the other bits -> 1011 0101 (signed magnitude)

1 = negative -> 011 0101 = magnitude

$(0x2^6) + (1x2^5) + (1x2^4) + (0x2^3) + (1x2^2) + (0x2^1) + (1x2^0)$

= (32 + 16 + 4 + 1)

= 53 -> magnitude -> add sign

= -53 (base 10)

(b) 1111 0010 -> keep msb -> invert the other bits -> 1000 1101 (signed magnitude)

1 = negative -> 000 1101 = magnitude

$(0x2^6) + (0x2^5) + (0x2^4) + (1x2^3) + (1x2^2) + (0x2^1) + (1x2^0)$

= 13 -> magnitude -> add sign

= -13 (base 10)

(c) 1000 0111 -> keep msb -> invert the other bits -> 1111 1000 (signed magnitude)

1 = negative -> 111 1000 = magnitude

$(1x2^6) + (1x2^5) + (1x2^4) + (1x2^3) + (0x2^2) + (0x2^1) + (0x2^0)$

= 120 -> magnitude -> add sign

= -120 (base 10)

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

(a) 1100 1010 -> subtract 1 to get 1’s comp -> 1100 1001 -> keep msb -> invert the other bits -> 1011 0110 (signed magnitude)

1 = negative -> 011 0110 = magnitude

$(0x2^6) + (1x2^5) + (1x2^4) + (0x2^3) + (1x2^2) + (1x2^1) + (0x2^0)$

= 54 -> add sign

= -54 (base 10)

(b) 1111 0010 -> subtract 1 to get 1’s comp -> 1111 0001 -> keep msb -> invert the other bits -> 1000 1110 (signed magnitude)

1 = negative -> 000 1110 = magnitude

$(0x2^6) + (0x2^5) + (0x2^4) + (1x2^3) + (1x2^2) + (1x2^1) + (0x2^0)$

= 14 -> add sign

= -14 (base 10)

(c) 1000 0111 -> subtract 1 to get 1’s comp -> 1000 0110 -> keep msb -> invert the other bits -> 1111 1001 (signed magnitude)

1 = negative -> 111 1001 = magnitude

$(1x2^6) + (1x2^5) + (1x2^4) + (1x2^3) + (0x2^2) + (0x2^1) + (1x2^0)$

= 121 -> add sign

= -121 (base 10)

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

b. -16_d

c. -21_d

d. -0_d

Each binary result represented in Signed magnitude representation.

(a) -100 -> start with the positive number -> |-100| = 100

100 / 2 = 50 Remainder 0

50 / 2 = 25 Remainder 0

25 / 2 = 12 Remainder 1

12 / 2 = 6 Remainder 0

6 / 2 = 3 Remainder 0

3 / 2 = 1 Remainder 1

1 / 2 = Remainder 1

100 = 0110 0100 -> since we need -100, the msb needs to be 1 -> **1110 0100**

(b) -16 -> start with the positive number -> |-16| = 16

16 / 2 = 8 Remainder 0

8 / 2 = 4 Remainder 0

4 / 2 = 2 Remainder 0

2 / 2 = 1 Remainder 0

1 / 2 = Remainder 1

16 = 0001 0000 -> since we need -16, the msb needs to be 1 -> **1001 0000**

(c) -21 -> start with the positive number -> |-21| = 21

21 / 2 = 10 Remainder 1

10 / 2 = 5 Remainder 0

5 / 2 = 2 Remainder 1

2 / 2 = 1 Remainder 0

1 / 2 = Remainder 1

21 = 0001 0101 -> since we need -21, the msb needs to 1 -> **1001 0101**

(d) -0 -> start with regular 0

0 / 2 = 0 Remainder 0

0 = 0000 0000 -> -0 needs to have an msb of 1 so -> **1000 0000**

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

(a) From 3.1 we got -100_d = 1110 0100 (signed magnitude)

From signed magnitude to 1's complement we keep the msb and invert other bits

1110 0100 -> **1001 1011**

(b) From 3.1 we got -16_d = 1001 0000 (signed magnitude)

From signed magnitude to 1's complement we keep the msb and invert other bits

1001 0000 -> **1110 1111**

(c) From 3.1 we got -21_d = 1001 0101

From signed magnitude to 1's complement we keep the msb and invert other bits

1001 0101 -> **1110 1010**

(d) From 3.1 we got -0_d = 1000 0000

From signed magnitude to 1's complement we keep the msb and invert other bits

1000 0000 -> **1111 1111**

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

(a) From 3.2 we got -100_d = 1001 1011 (1's complement)

To get 2's complement we add 1 to the 1's complement

1001 1011 -> **1001 1100**

(b) From 3.2 we got -16_d = 1110 1111 (1's complement)

To get 2's complement we add 1 to the 1's complement

1110 1111 -> **1111 0000**

(c) From 3.2 we got -21_d = 1110 1010 (1's complement)

To get 2's complement we add 1 to the 1's complement

1110 1010 -> **1110 1011**

(d) From 3.2 we got -0_d = 1111 1111 (1's complement)

To get 2's complement we add 1 to the 1's complement

1111 1111 -> **0000 0000**

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

A. An unsigned 7-bit number?

0 to $(2^7 - 1)$ -> **0 to +127**

B. A signed 7-bit number?

$-(2^{7-1})$ to $(2^{7-1} - 1)$ -> **-64 to +63**

5. (12 points) Provide the answer to the following problems ($\wedge$ = AND, $\vee$ = OR)

1. 1000 $\wedge$ 1110

$$\begin{array}{r} 1\,0\,0\,0 \\ \underline{\wedge\, 1\,1\,1\,0} \\ \mathbf{1\,0\,0\,0} \end{array}$$

2. 1000$\vee$1110

$$\begin{array}{r} 1\,0\,0\,0 \\ \underline{\vee\, 1\,1\,1\,0} \\ \mathbf{1\,1\,1\,0} \end{array}$$

3. (1000∧1110) ∨ (1001∧1110)

$$\begin{array}{r} 1\,0\,0\,0 \\ \underline{\wedge\,1\,1\,1\,0} \\ 1\,0\,0\,0 \end{array} \qquad \begin{array}{r} 1\,0\,0\,1 \\ \underline{\wedge\,1\,1\,1\,0} \\ 1\,0\,0\,0 \end{array}$$

$$\begin{array}{r} 1\,0\,0\,0 \\ \underline{\vee\,1\,0\,0\,0} \\ \mathbf{1\,0\,0\,0} \end{array}$$

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

<u>- 65</u> (5 – 5 = 0, and 2 – 6 = - 4)

- 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.

a. Convert 25 to binary

25 / 2 = 12 Remainder 1

12 / 2 = 6 Remainder 0

6 / 2 = 3 Remainder 0

3 / 2 = 1 Remainder 1

1 / 2 = Remainder 1

25_d = 0001 1001

* Since 25 is positive, its binary form is the same in two's complement

b. Convert -65 to binary

start with the positive number 65

65 / 2 = 32 Remainder 1

32 / 2 = 16 Remainder 0

16 / 2 = 8 Remainder 0

8 / 2 = 4 Remainder 0

4 / 2 = 2 Remainder 0

2 / 2 = 1 Remainder 0

1 / 2 = Remainder 1

65_d = 0100 0001 (unsigned)

c. Convert to signed binary -> then 1's comp -> then 2's comp

We only need to convert -65_d into 2's complement

Put a 1 in the msb to make 65 into -65

0100 0001 -> 1100 0001 (signed magnitude)

Keep msb and invert the other bits to get 1's comp

1100 0001 -> 1011 1110 (1's comp)

Add 1 to the 1's comp to get 2's comp

1011 1110 -> 1011 1111 (2's comp)

d. Add the binary/2's complement numbers

0001 1001

+ 1011 1111

1101 1000 (2's comp)

e. Convert -40_d from answer 6 to 2's comp

-40 -> start with the positive number -> |-40| = 40

40 / 2 = 20 Remainder 0

20 / 2 = 10 Remainder 0

10 / 2 = 5 Remainder 0

5 / 2 = 2 Remainder 1

2 / 2 = 1 Remainder 0

1 / 2 = Remainder 1

40 -> 0010 1000 (unsigned)

Add a 1 to the msb to go from 40 to -40

0010 1000 -> 1010 1000 (signed magnitude)

Keep the msb and invert the other bits to get 1's comp

1010 1000 -> 1101 0111 (1's comp)

Add 1 to 1's comp to get 2's comp

1101 0111 + 1 -> **1101 1000 (2's comp)**

f. From part d the answer is the same as -40 from part e. So the answer is correct.