Homework 2

1. Unsigned: 0, 13, 24, 63
Signed: 16, -2, 31, -32

Unsigned:

For each decimal number, we convert it directly to binary.

- Decimal number 0 in binary is 000000
- Decimal number 13 in binary is 001101 : $2^3+2^2+2^0=13$
- Decimal number 24 in binary is 011000 : $2^4+2^3=24$
- Decimal number 63 in binary is 111111 : $2^5+2^4+2^3+2^2+2^1+2^0=63$

Signed:

For positive numbers, we also convert them directly to binary using base 2.

- Decimal number 16 in binary is 010000 : $2^4=16$ — for positive num.
- Decimal number -2 in binary is 111110 — for negative — $2=-2$ — $(2-1)=000001 \rightarrow$ inverse $=$ 111110
- Decimal number 31 in binary is 011111 — for postive num. — $31=2^4+2^3+2^2+2^1+2^0$
- Decimal number -32 in binary is 100000 — for negative — $-32 = 32-1 = 31 = 011111 \xrightarrow{\text{Inverse}} \rightarrow 100000$

2. Formula:

Basically, check the first bit, If it is 1 $\Rightarrow$ $\Rightarrow$ number negative; otherwise it is unsigned, then If 1 then find the two's complement by inverting all the bits (0->1; 1->0) and add 1. Then convert result to the decimal

Then we have

1. 000101
   - $\to$ Signed: $5 = 2^2 + 2^0 = 5$
   - $\to$ Unsigned $5 = 2^0 + 2^2 = 5$

2. 101011
   - $\to$ Signed: $-21 = -2^5 + 2^3 + 2^1 + 2^0$
   - $\to$ unsigned: $43 = 2^5 + 2^3 + 2^1 + 2^0 = 43$

3. 111111
   - $\to$ signed: -1 $\quad -2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$
   - $\to$ unsigned: 63 $\quad 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 = 63$

4. 100000
   - $\to$ Signed: -32 $\quad -2^5 = -32$
   - $\to$ unsigned: $32 = 2^5$

3. $7 = 16^0 \cdot 7 = 0x07$

$171 = 16^1 \cdot 10 + 11 \cdot 16^0 = 0xAB$

$240 = 16^1 \cdot 15 = 0xF0$

$126 = 16^1 \cdot 7 + 16^0 \cdot 14 = 0x7E$

4. 0x3C = 00111100

0x7E = 01111110

0xFF = 11111111

0xA5 = 10100101

5. $00111100 \xrightarrow{\text{negation}} 11000011$

$01111110 \xrightarrow{\text{negation}} 10000001$

$11111111 \xrightarrow{\text{negation}} 00000000$

$10100101 \xrightarrow{\text{negation}} 01011010$

6. 0x DEADBEEF

For Big Endian:

0xDE	0xAD	0xBE	0xEF

For Little Endian:

0xEF	0xBE	0xAD	0xDE

Basically, the order of the bytes in memory is reversed between the big-Endian and little-Endian conventions.

7. $7 = 00111$

$15 = 01111$

$-16 = 10000$

$-5 = 11011$

Sign-extension:

$7 = 00000111$

$15 = 00001111$

$-16 = 11110000$

$-5 = 11111011$

Zero-extension:

$7 = 00000111$

$15 = 00001111$

$-16 = 00010000$

$-5 = 00011011$

8. 1. $7 = 2^2 + 2^1 + 2^0 = 0111$

$9 = 2^3 + 2^0 = 1001$

$$\begin{array}{r} 0111 \\ 1001 \\ \hline 10000 \end{array}$$

→ we got overflow and the result will be $0000 = 0$

2. $4_{10} = 0100$

$-5 = 1011$

$$\begin{array}{r} 0100 \\ 1011 \\ \hline 1111 \end{array}$$

→ we don't have overflow and the result will be $1111 = -1$