**Given the bitstring (11 0011)2, what decimal number does it represent?**

**It depends!**

**If unsigned:**

**1\*2^5 + 1\*2^4 + 0\*2^3 + 0\*2^2 + 1\*2^1 + 1\*2^0**

**32+16+0+0+2+1**

**51**

**If signed: It depends!**

**If Sign-Magnitude:** The most significant bit (MSB) is the sign, the rest is the magnitude.

**1 10011**

**- 1\*2^4 + 0 + 0 + 1\*2^1 + 1\*2^0**

**- 16 + 2 + 1**

**-19**

**0 10011**

**+ 1\*2^4 + 0 + 0 + 1\*2^1 + 1\*2^0**

**+ 16 + 2 + 1**

**19**

**If 1’s Complement:**

**110011**

**For 1’s complement numbers, check the sign first. If negative, perform a bitwise inverse on the bitstring.**

**MSB gives us the sign – 110011 is negative**

**bitwise inverse: 001100**

**-12**

**What about 011100?**

**For 1’s complement numbers, check the sign first. If positive, do not make any changes to the number.**

**011100 = 16+8+4 = 28**

**if 2’s Complement:**

**110011**

**For 2’s complement numbers, check the sign first. If negative, perform a bitwise inverse on the bitstring then add 1.**

**MSB gives us the sign – 110011 is negative**

**bitwise inverse: 001100**

**add 1:**

**001100**

**000001**

**= 001101**

**-1 \* (8+4+1)**

**-13**

**What about 011100?**

**For 2’s complement numbers, check the sign first. If positive, do not make any changes to the number.**

**011100 = 16+8+4 = 28**

floats/doubles: IEEE 754 standard for FP values

integers: sign bit / value

floats/doubles: sign bit, exponent, mantissa

Let’s start with a decimal (base 10 number)

37

unsigned: 100101

37 = 2 \* 18 + 1

18 = 2 \* 9 + 0

9 = 2 \* 4 + 1

4 = 2 \* 2 + 0

2 = 2 \* 1 + 0

1 = 2 \* 0 + 1

What about number of bits?

int 0000 0000 0000 0000 0000 0000 0010 0101

short 0000 0000 0010 0101

long 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0010 0101

-42

42 = 32 + 8 + 2

00101010

Sign-Extend to 16 bits:

0000 0000 0010 1010

Sign-Magnitude:

1000 0000 0010 1010

1’s Complement:

v1 00101010

11010101 bitwise inverse

1111 1111 11010101 sign extend to 16 bits

v2 0000 0000 0010 1010 starts with 16 bits

1111 1111 1101 0101 bitwise inverse

2’s Complement of -42

v1 00101010 start with 42

11010101 bitwise inverse

00000001 add 1

11010110 result

1111 1111 1101 0110 sign extend to 16 bits

v2 00101010 start with 42

0000 0000 0010 1010 sign extend to 16 bits

1111 1111 1101 0101 bitwise inverse

0000 0000 0000 0001 add 1

1111 1111 1101 0110 result

(-4)0100 -> 1011 -> 1011 + 1 = 1100

(-5)0101 -> 1010 -> 1010 + 1 = 1011

1100 -> 0011 -> 0011 + 1 = 0100

1110 -> 0001 -> 0001 + 1 = 0010 (-2)