Lecture 6:
Integers: Sign and Size (cont.)

CSE 29: Systems Programming and Software Tools

Aaron Schulman (Shalev)

UCSDCSE
Computer Science and Engineering

Today’s Lecture

- How do we use different sized integers in C
- How humans read binary values in computers
- Project discussion

Two’s Complement

- What if we make the MSB equal to -2^{MSB}?
  - In other words, if the MSB is set, the number becomes negative with that magnitude

MSB **LSB**

MSB							LSB
-128	64	32	16	8	4	2	1
-2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

- Minimum will have higher magnitude than the maximum (by -1)
  - Min (-128)
  - Max (127 = 64 + 32 +16 +8 + 4 + 2 + 1)

- Only one zero, and hardware is the same as an unsigned int!

Data types in typically used in C

- Integer data types
  - **char** = ‘A’ (1 byte – max 127) - *Signed*
  - **int/int32_t** = 42 (4 bytes – max 2 billion) - *Signed*
  - **unsigned char** = (1 byte – max 255) - *Unsigned*
  - **unsigned int/uint32_t** = (4 bytes - max 4 billion)
  - long long int/int64_t = (8 bytes – max 8 quad…)
  - unsigned long long int/uint64_t =
    (8 bytes – max 16 quad…)

Integers in Computers today

Integers and Addresses: 64 bits (8 bytes) – “Word” Size

00000000	**00000000**	**00000000**	**00000000**	**00000000**	**00000000**	**00000000**	**00000000**
1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte	1 byte

1 integer = 64 bits (~18 quintillion)

Older computers used 32 bit words:
2^{32} = 1024 * 1024 * 1024 * 4 = 4 billion

Humans struggle with reading binary

What is this binary data?

1111110001011011011111111001101101110100010111100101111010000101

In binary representation 64 bits = 64 numbers to read ☹

What if we represent the binary data in base 16 (2^4 possibilities per digit)

0,1,2,3,4,5,6,7,8,9, A *(10)*, B *(11)*, C *(12)*, D *(13)*, E *(14)*, F *(15)*

0xFC5B7F9B745E5E85

Hex helps humans read binary

Hexadecimal notation divides binary data into 4-bit groups:

1111110001011011011111111001101101110100010111100101111010000101

0xFC5B7F9B745E5E85

1111	1100	0101	1011	0111	1111	1001	1011	0111	0100	0101	1110	0101	1110	1000	0101
F	C	5	B	7	F	9	B	7	4	5	E	5	E	8	5

Hex is easier for humans to read & easier for humans to say (kind of like why we divide phone numbers into groups)

Hex helps humans read binary

What is this hexadecimal number in binary?

0xFF

1111 1111

0x18

0001 1000