CS 354
Machine Organization and Programming
Lecture 10

Michael Doescher
Summer 2020

Low Level C Programming
Number Representations

Number Representations

Signed Magnitude
One’s Complement
Two’s Complement

What is the min/max integer we could represent using these representations

Number Representations

Signed Magnitude
One's Complement
Two's Complement

What is the min/max integer we could represent using these representations

How to work with this in C?

- C uses two's complement representation
- Enter a specific bit pattern
- How printf addresses these
- sizeof
- Min/max

Type Casting

```
char  -> short
short -> int
int   -> long
long  -> long long
```

Type Casting

```
char  -> short       1 byte  -> 2 bytes
short -> int         2 bytes -> 4 bytes
int   -> long        4 bytes -> 4 bytes
long  -> long long   4 bytes -> 8 bytes
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x??
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
0x1A       : 0x001A
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
0x1A        : 0x001A
0001 1010   : 0000 0000 0001 1010
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
0x1A        : 0x001A
0001 1010   : 0000 0000 0001 1010

Negative numbers
Decimal -26
Binary = ???  (1 byte : 2 bytes)
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
0x1A        : 0x001A
0001 1010   : 0000 0000 0001 1010

Negative numbers
Decimal -26
Positive 26      : 0001 1010
Ones complement  : 1110 0101
Add 1            : 1110 0110
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
0x1A        : 0x001A
0001 1010   : 0000 0000 0001 1010

Negative numbers
Decimal -26
1110 0110   : 1111 1111 1110 0110
```

Type Casting

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
0x1A        : 0x001A
0001 1010   : 0000 0000 0001 1010

Negative numbers
Decimal -26
1110 0110   : 1111 1111 1110 0110
0xE6        : 0xFFE6
```

Sign Extension

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 26 = 0x1A
0x1A          : 0x001A
0001 1010     : 0000 0000 0001 1010

Negative numbers
Decimal -26
1110 0110     : 1111 1111 1110 0110
0xE6          : 0xFFE6
```

Sign Extension

```
char  -> short

Positive numbers (signed or unsigned)
Decimal 1 = 0x01
0x01          : 0x0001
0000 0001     : 0000 0000 0000 0001

Negative numbers
Decimal -1
1111 1111     : 1111 1111 1111 1111
0xFF          : 0xFFFF
```

Type Casting

```
short -> char  : If it fits (less than CHAR_MAX)

Positive numbers (signed or unsigned)
Decimal 0026 = 0x001A
0x001A         : 0x1A
0000 0000 0001 1010  : 0001 1010

Negative numbers
Decimal -26
1111 1111 1110 0110  : 1110 0110
0xFFE6         : 0xE6
```

Type Casting

```
short -> char  : If it doesn't fit
(greater than CHAR_MAX)

Positive numbers (signed or unsigned)
Hex 0x1234 = decimal ????

0x1234         : 0x??
0001 0010 0011 0100  : ???? ????
```

Type Casting

```
short -> char  : If it doesn't fit
(greater than CHAR_MAX)

Positive numbers (signed or unsigned)
Hex 0x1234 = decimal 4660

0x1234         : 0x34
0001 0010 0011 0100  : 0011 0100
```

Type Casting : Unsigned Types

```
Sign Extension doesn’t happen for unsigned types!!!

char -> short
0000 0001 -> 0000 0000 0000 0001
1111 1111 -> 0000 0000 1111 1111
```