CS354: Lec 002

Number Systems

- Lowest-level representation of data is called a *bit*; can be either 0 or 1.
- A *byte* is 8 of these bits, for example: `10100100`
- The leftmost digit is the *most significant bit* or MSB, and the rightmost digit is the *least significant bit* or LSB.
- With just two bits, we can represent 4 numbers: `01, 10, 00, 11`.
    - With this pattern we can find how many numbers that can be represented with a certain amount of binary digits with the expression below, where x is the amount of digits and y is the amount of numbers that can be represented.

$y = 2^x$

- Since binary is a bit annoying to write, we use hex, which converts 4 binary digits to 1 character. As such, `1011 0100 1011 1001 1010 1011 1010 1111` is converted to `B4B9ABAF`, or conventionally, `0xB4B9ABAF`. The `0x` prefix is used to tell the computer that the characters after it will be written in hexadecimal. A demo is shown below:

```c
#include<stdio.h>
int main() {
    int num = 0xB4B9ABAF;
    printf("num = %x\n", num);
    printf("num = %d\n", num);
    return 0;
}
// output
num = b4b9abaf
num = -1262900305
```

Memory Model

A memory is made up of a certain amount of bytes. Say you have a memory made up of 16 bytes, each byte taking up 1 slot in the total memory.

- Suppose you store a 4 byte int `0x0000000B` at address 4. Since the entire int is 4 bytes, it cannot all be stored in one byte slot. Instead, the value is stored from addresses 4-7.

More Basic C Stuff

- A `char` is 1 byte. You can map it to a binary representation, and can therefore do math on it. For example, if you set char to `A` and do `char = char + 1`, `char` is now `B`.
- A `double` is always more accurate than a `float`.
- `int` is typically 4 bytes.
- `short` is typically 2 bytes.
- `long` is typically 8 bytes.
- C does not contain strings or boolean values, at least as a primitive type. Instead, we can create a string using a character array, e.g. `char course[] = "CS354"`. This will be constructed as `C` `S` `3` `5` `4` `\0` (the last character is the null terminal character which terminates all strings).
- Operators are identical to java.
- Booleans can be represented by numbers: 0 is false, and everything else is true.
- C has `while` loops, `for` loops, and `do-while` loops.