Bytes and Hex

CS 1428
Fall 2011

Jill Seaman

Lecture 16

Computer memory

- Memory is a sequence of bits:
  - bit is on or off
  - represent with 0 or 1
- 1 byte = 8 bits
- char is stored in 1 byte
- sizeof(x) returns the size of data in bytes

Measuring computer memory

- Memory is measured in bytes
- using powers of 2

1 Kilobyte = 2^{10} =	1,024 bytes	about a thousand
1 Megabyte = 2^{20} =	1,048,576 bytes	about a million
1 Gigabyte = 2^{30} =	1,073,741,824 bytes	about a billion
1 Terabyte = 2^{40} =	1,099,511,627,776 bytes	about a trillion

How big are files?

- Some typical sizes of files containing certain data:

pdf file: about 200KB (Lecture15.pdf=193KB)

photo: about 500KB - 1.4MB

song: about 3.5MB to 8MB

video: 5 min: 92 MB
full length movie 600-700MB up to around 2GB

How much memory is in ...?

- Some typical sizes of storage in various devices

ipod: 2GB (shuffle) 160GB (classic)

iphone: 8/16/32/64 GB

RAM in a computer: 1 to 4GB (depending on age)

Hard drive in laptop: 120GB, 500GB, 750GB
Hard drive in desktop: 500GB, 1TB

External hard drive: 320GB, 3TB

How many songs can I store?

- If one MP3 song is 20MB, how many songs will fit on a 4.7GB DVD?

```
How many times 20MB fits into 4700MB:

x * 20MB = 4.7GB

1GB = 1000MB, so multiply rhs by 1000:

x * 20MB = 4700MB

x = 4700/20 = 235
```

Memory Addresses

- Every byte in Main Memory (Ram) has an "address".
- The address is a number
- The locations (bytes) are numbered in sequential order:

  0, 1, 2, 3, 4, 5, 6, . . . several million or billion

Memory Addresses

- You can use the "address of" operator to find the address of any variable in your program:

```
int main () {

   int x;
   cout << &x << endl;

}
```

```
Output: 0xf79c14
```

- What is "0xf79c14"?

Hexadecimal

- Number systems:
  - binary: (base 2) 0,1
  - decimal: (base 10) 0,1,2,3,4,5,6,7,8,9
  - hexadecimal: (base 16) 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
- In C++, hexadecimal values have “0x” in front of them (that is “zero x”).
- Not enough digits: Use A for 10, B for 11, C for 12, D for 13, E for 14, and F for 15.

Counting in various systems

Decimal	Binary	Hexadecimal	
0	0	0	
1	1	1	
2	10	2	
3	11	3	
4	100	4	
5	101	5	
6	110	6	
7	111	7	
8	1000	8	
9	1001	9	
10	1010	A	
11	1011	B	
12	1100	C	
13	1101	D	
14	1110	E	
15	1111	F	
16	10000	10	
17	10001	11	...

Converting from hex to binary

- Hex to binary:
  - Replace each hex digit with its 4-bit binary equivalent (pad 1,2,3, bit values with zeros).

A3 = 1010 0011

2E9A = 0010 1110 1001 1010

Converting from binary to hex

- Binary to hex:
  - break the binary up into 4 bit segments (start from the right).
  - Replace each 4-bit segment with the corresponding hex digit from table:

0110110000 => 0001 1011 0000
(pad the left with zeros)

=> 1 B 0

Back to the memory address:

- What is “0xf79c14”?
  - f79c14 is in hexadecimal

F79C14 = 1111 0111 1001 1100 0001 0100

$15x16^5 + 7x16^4 + 9x16^3 + 12x16^2 + 1x16^1 + 4x16^0$

= 16,227,348