Primitive Data Types

integers

byte	from -128 to 127 (8 bits = 1 byte)
short	from -32,768 to 32,767 (16 bits = 2 bytes)
int	from -2,147,483,648 to 2,147,483,647 (32 bits = 4 bytes)
long	from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 (64 bits = 8 bytes)

real numbers

float	real numbers (with fractional components)
double	more precise real numbers

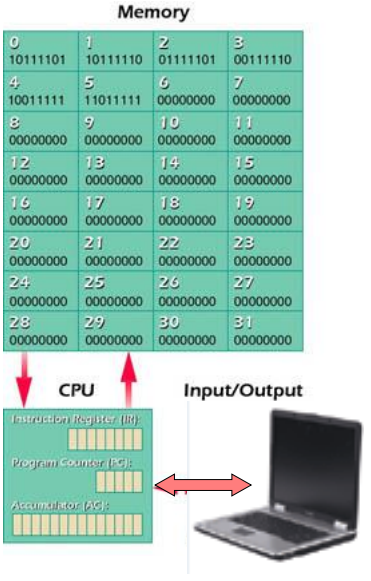
other

boolean	true or false values (only 1 bit)
char	character (ASCII) (8 bits = 1 byte).

Main Memory (RAM)

- Is organized as a sequence of locations
- Each location can contain a sequence of bits, representing data or an instruction
- Locations are numbered with "addresses" from 0 to memory-size

(can be huge, even gigabytes)



Why use binary numbers?

- We could build computers that operate in base-10, but they would be expensive.
- What is easier than having a switch be ON or OFF?
- Electricity is used… that way: low current or high current.
- High voltages indicate 1 and low voltages indicate 0

Data Representation: Bits

The bit is the smallest unit of data; it is 1 or 0, that is, on or off

Base Ten Numbers

- In base 10, a *decimal integer 587* means:

$$587 = 5 \times 10^2 + 8 \times 10^1 + 7 \times 10^0$$

$7 \times 10^0 = 7$

$8 \times 10^1 = 80$

$5 \times 10^2 = 500$

587

Base Two

- Likewise, in base 2, a binary integer 101 means:

$$101 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

$1 \times 2^0 = 1$

$0 \times 2^1 = 0$

$1 \times 2^2 = 4$

5

Convert the binary (base two) number 110111 to decimal (base ten).
Answer = 55

0	0	1	1	0	1	1	1
x 128	x 64	x 32	x 16	x 8	x 4	x 2	x 1
+ 0	+ 0	+ 32	+ 16	+ 0	+ 4	+ 2	+ 1 = 55

Convert the decimal (base ten) number 52 to binary (base two).
Answer = 00110100

	x 32	x 16	x 8	x 4	x 2	x 1
52	1	1	0	1	0	0
-32						
20						
-16						
4						
-4						
0						

How do we represent negative numbers
Two's Complement

13 =	0 0 0 0 1 1 0 1	
	1 1 1 1 0 0 1 0	*invert the digits*
	1	*add 1*
-13 =	1 1 1 1 0 0 1 1	

http://www.cs.cornell.edu/~tomf/notes/cps104/twoscomp.html

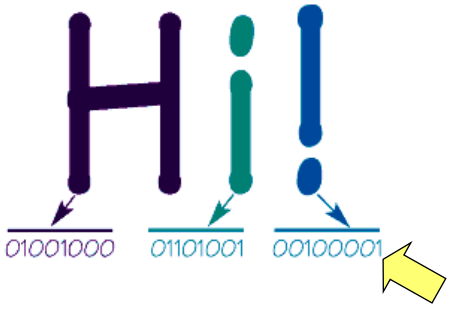
Characters, marks, and more

Letters, numbers, line feeds, and non-printing characters

ASCII-8. Pronounced "ass-key" ASCII stands for *A*merican *S*tandard *C*ode for *I*nformation *I*nterchange, has 256 different symbols-all Operating Systems fully understand ASCII.

UNICODE allows for up to 65,536 different characters. It is more complex and not implemented on many Operating Systems, but it is on Windows NT and Windows XP.

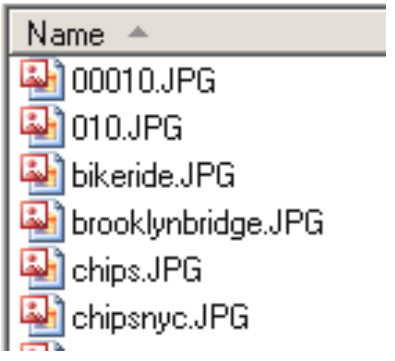
Different characters held in RAM



00100001 = ???

Why are these lists in alphabetical order?

prod_name
12 inch teddy bear
18 inch teddy bear
8 inch teddy bear
9 inch nails
Bird bean bag toy
Fish bean bag toy
King doll
Queen doll
Rabbit bean bag toy



ASCII

ASCII-8 represents 256 characters -- the foreign language possibilities are below:

Symbol	Decimal	Binary
A	65	01000001
B	66	01000010
C	67	01000011
D	68	01000100
E	69	01000101
F	70	01000110

8	BS
9	TAB
10	LF
11	VT

134	å	150	û	166	ª	182	╢	199	╟	213	
135	ç	151	ù	167	º	183	╖	200	╚	216	
136	ê	152	_	168	¿	184	╕	201	╔	217	
137	ë	153	Ö	169		185	╣	202	╩	218	

ASCII, Cont.

- 32 is the ASCII code for a space.
- So 32 = 00100000 in binary, and when the computer gets that data, it causes a space to appear.
- Note: all the capital letters finish before the lower case letters appear

- B = 66
- b = 98

65	A
66	B
67	C
68	D
69	E

97	a
98	b
99	c
100	d
101	e
102	f