how computers work (1)

more on representing information
digitally

overview

- already seen: decimal numbers, pictures
  can be represented using bits (0's and 1's)

- today:
  - using bits to represent text,
    keyboard inputs
  - subtleties of number representation

symbols represent information in many contexts

- dots and dashes in morse code

- raised dots are used to represent symbols
  in braille

- nucleotides (A,C,G,T) in our DNA encode
  our genetic makeup

- bits represent then information stored in a
  computer's memory

why bits?

- electrical signals used to transmit information
  can be **on** or **off**

- data storage media such as disks have positions
  that are **positively** or **negatively** charged

- two possibilities – on/off, +ve/-ve – are
  represented as **0** and **1**

- (there is nothing particularly special about 0 and
  1, any pair of symbols would work just as well)

digitizing text, keyboard inputs

- text contains the symbols we see: letters,
  numbers, punctuation marks, spaces, and
  other symbols (e.g. for arithmetic, business)

- text also contains "nonprintable" characters:
  new-line, tab

- keyboards have yet more symbols on their
  keys, such as backspace, function keys

ASCII standard (American Standard Code for Information Interchange, 1960's)

- the ASCII standard uses bit strings of length 8
  to represent symbols

- see ASCII standard table in text, Chapter 8

- for example, "A" is represented as "01000001",
  and the symbol "@" is represented as
  "01000000"

- how many symbols in total can be represented
  with 8 bits?

representing text in binary

symbol	bit representation
a	01100001
b	01100010
c	01100011
d	01100100

- "bad" is represented as "01100010 011000101100100"
- what does "011000110110000101100010" represent?

decimal digits in ASCII

- decimal digits also have ASCII representations

- for example, my office phone number, "822 1964", is represented as "00111000 00110010 00110010 00110001 00111001 00110110 00110100"

- decimal numbers embedded in text are represented in ASCII, *not* by converting to their binary representation

confusion with number representation

- when doing computer arithmetic, it is appropriate to represent numbers using binary notation

- when treating numbers as symbols (e.g., phone number), it is appropriate to represent the digits using ASCII notation

  When using numbers in a program, it can be important to know which representation is used!

example: numbers in JavaScript

- suppose 'confuse' is declared as a variable:

  var confuse;

- depending on how 'confuse' is used in the code, it may store a number as a text string, or as an integer

numbers in JavaScript

- suppose **confuse** has value **1**
- if **confuse** is a string, **confuse + 2 + 3** is the string **"123"**
- if confuse is an integer, then **confuse + 2 + 3** is **6**!
- to help programmers, JavaScript provides ways to convert from text format to integer format
- if **confuse** is a string, then **parseInt(confuse) + 2 + 3** is **6**

numbers in java

- java syntax is designed to avoid any confusion about the representation of numbers

- the words **int** and **String** are used in variable declarations, to declare right from the start which type the variable has

  Examples:
    int width;
    String userName;

resources

- text chapter 8 (required reading)

- A Brief History of Character Codes,
  http://tronweb.super-nova.co.jp/characcodehist.html
  (useful additional reading)

- an on-line ASCII table + some additional
  information: www.jimprice.com/jim-asc.htm

- an on-line ASCII converter:
  http://nickciske.com/tools/binary.php