Computer Science
North University of China

CS101 Fall 2017

Introduction to
Computer Science

Instructor: Qingsong Guo
School of Computer Science & Technology

http://abelgo.cn/cs101.html

Lecture 2:
Welcome to the Digital World !

Think in Data: An Overview

Physical World, Models, & Data Representation

Computer as An Data Manipulator

Black box model

Data manipulator

Computer System: Hardware + Software

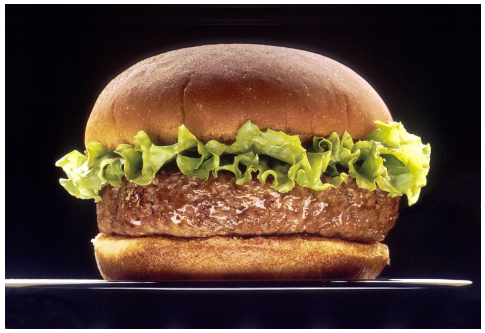
Computer is a sort of hardware and software that transforms, ingests and manipulates, the input data into output data in the form of our demand.

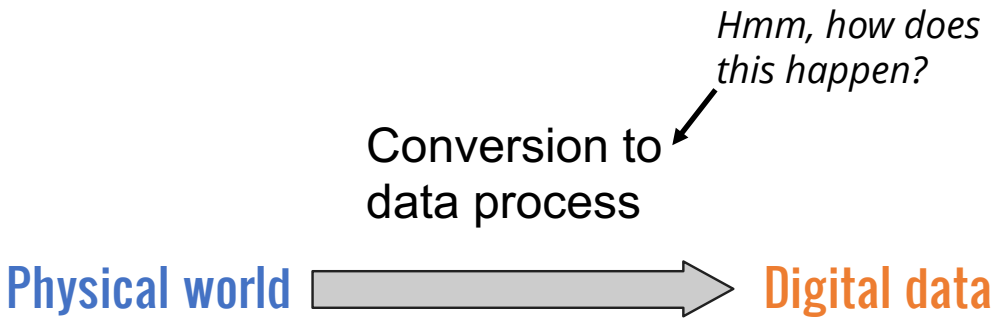
Computers and Data

A computer can only perform computation on digital data

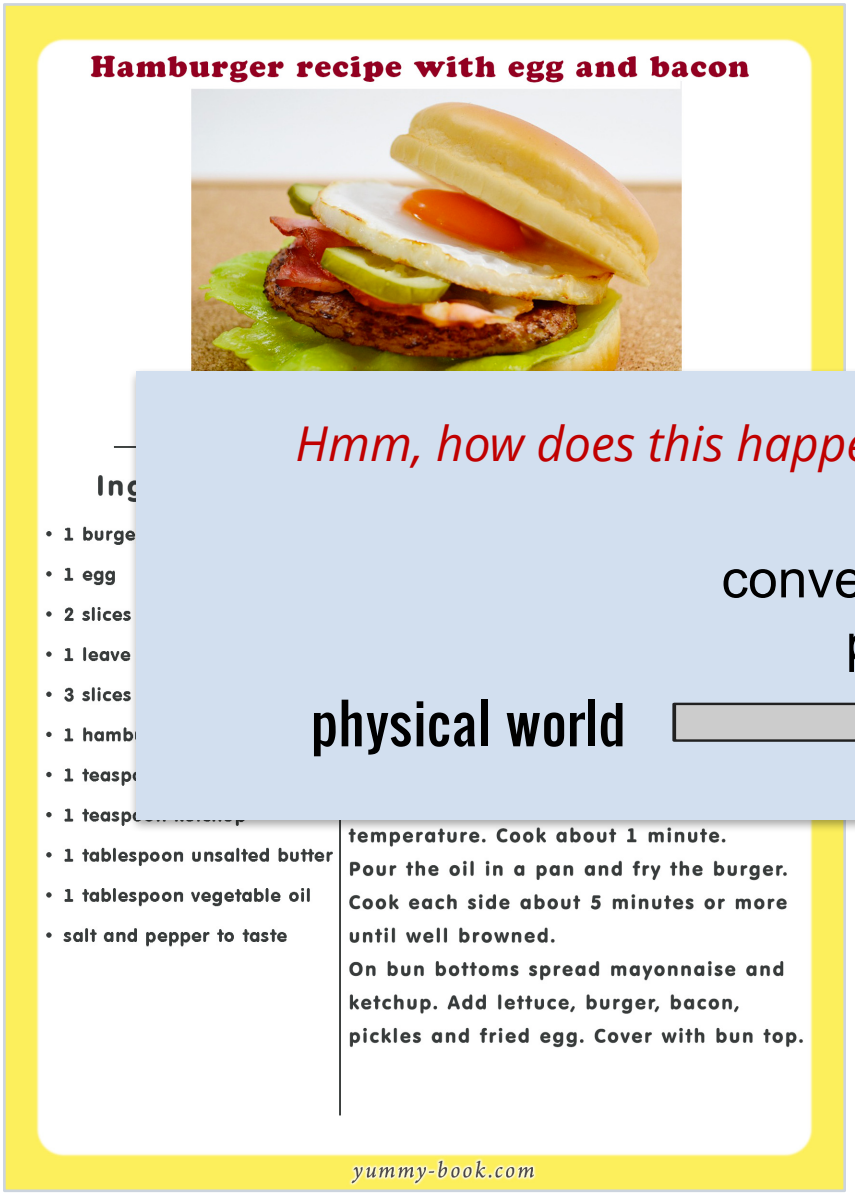
- It cannot directly perform computation on the real, physical world
- The physical world must first be converted into data
- A computer thus performs computation on a representation of the physical world

But, how, exactly, does one convert the real world into data?





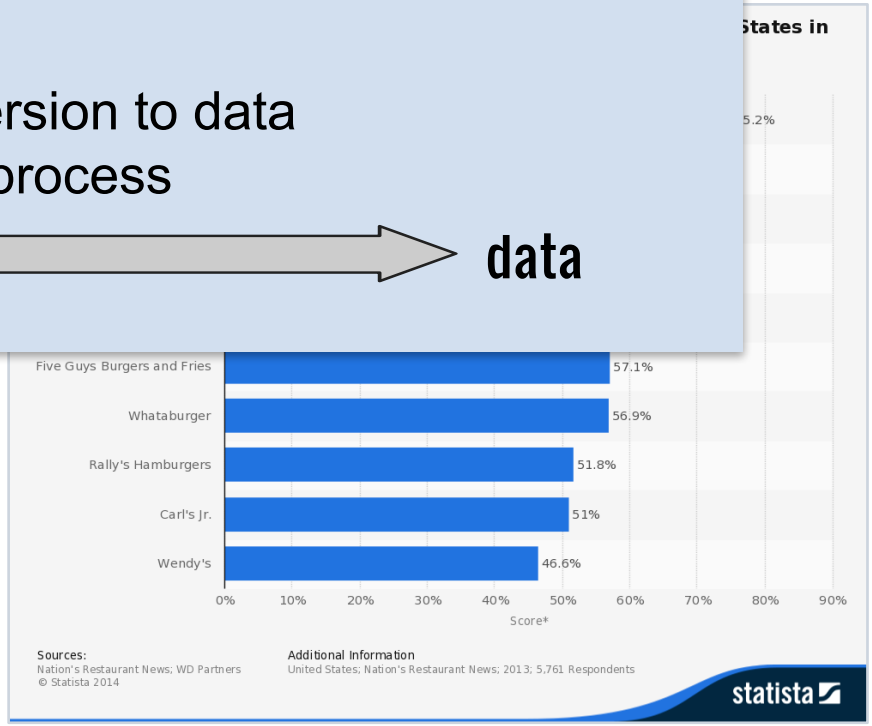
Data Related to Hamburger



Hmm, how does this happen?

conversion to data process

physical world → data



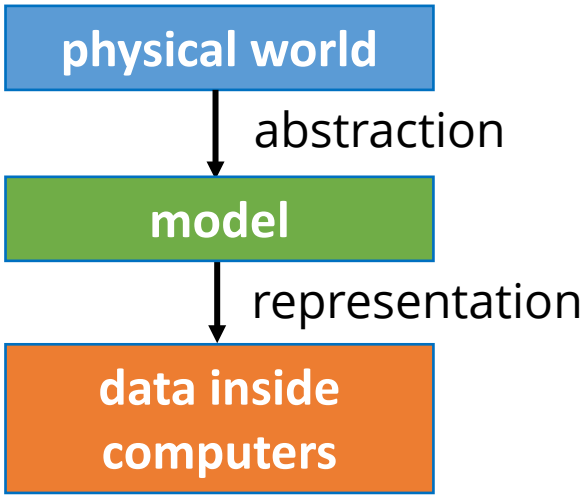
Abstraction and Models

Converting the real world into data

- Create a model of the real world
- Represent that model in data

How do you model the real world?

- Involves a process called abstraction



Abstraction

- Prerequisite: know your problem or application
- Focus on aspects of the real world that are important to the problem and add those elements to your model
- Omit elements of the real world that aren't relevant
- Implies: the same real world scenario can be modeled in many ways, depending on the problem at hand

Modeling Superman

Let's model the very first Superman comic book (漫画)

First, what is our problem?

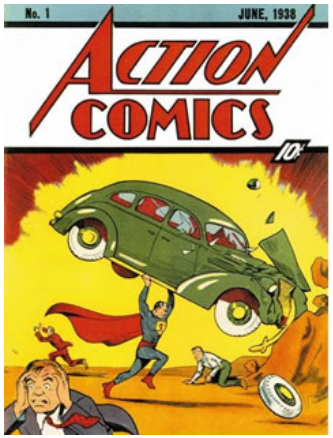
- How about, reading the comic on a computer screen?
- It is important to see the text, pictures, and page layout
- So, these must be in the model
- We can model the comic book as a set of page images
- Can represent the page images using the Portable Document Format (PDF)
- Or, alternately, could represent as a series of PNG, JPEG, or GIF images

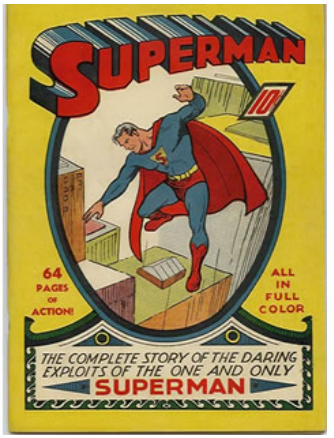
Modeling Superman (cont.)

New problem: have the computer speak aloud the dialog in the comic strip (连载漫画)

- It is important to have the text of the dialog so that the computer can convert it into voice
- Not important: actual images of the comic (let's assume the reader/listener has the comic in front of them)
- We model the comic strip as a series of frames, each containing the dialog of the characters
- We represent this model as a list of frames, where each frame has text that represents the dialog said by each character

Frame 1
Superman: "I will get you, evil dude!"
Evil Dude: "Eat kryptonite, caped spandex-boy"





Modeling Superman: observations

From the same physical situation (a comic book) have two separate models

- Page-focused model: for problem of reading on computer
  - Emphasizes images over machine-readability of dialog text
- Dialog-focused model: for problem of reading dialog out loud
  - Emphasizes machine readability, not images at all

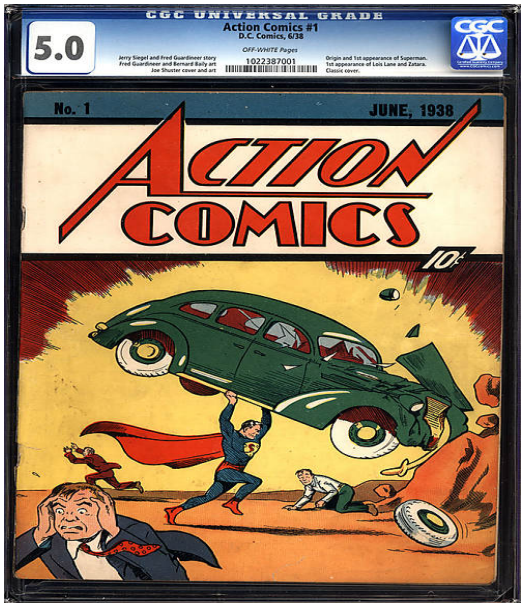
But, recall that the model is not the same as the real, physical system

- It is just an abstraction of the physical system

A model is not reality

A real story

- In summer 2010, a family was cleaning out their house, just prior to moving out due to foreclosure (抵押)
- They found a copy of Action Comics #1, condition VG/F
- By selling the comic, they raised $436,000, enough money to save the house
- www.comicconnect.com/bookDetail.php?id=355202



Reality

- Sometimes there is no substitute for the real, physical system
- Many aspects cannot be reproduced inside a computer
- Feel and smell of the paper
- Ownership of the physical comic as a status token

Importance of Standards

One challenge of modeling real-world systems:

- Ensuring that multiple groups all model the same situation in the same way, and
- Represent these models in the same way
- Computer systems often take input data from many sources, need to ensure data means the same thing in all locations

Importance of Standards (Cont.)

Consider temperature

Simple problem: units

- One site measures in Beijing, another in Shanghai
- Oh that would never happen: Mars Climate Orbiter
- Flight system software expected thrust in units of Newtons, while ground crew computed using units of Pound-force

Complex problem: measurement process

- One site measures in sun, next to a parking lot, using a digital thermometer
- Another site measures in shade, in a forest, using a mercury thermometer
- Oh that would never happen: Climate denial debate
- One source of climate denial concerns methods of collecting temperature data

Summary

Modeling the physical word

- Through the process of abstraction, we make models of the physical world.
- Models are represented as digital data
- Computers can operate on data

But, how to represent models as data?

Representing Models as Data

Basic data types (this lecture)

- Integers
- Floating point
- Boolean
- Characters
- Strings

Basic data structures (future lecture)

- Lists
- Arrays
- Stacks
- Trees
- Graphs

Representing Numbers

Today, two broad ways to represent numbers

Integers

- A direct mapping of a number into binary digits
- Very precise, can exactly represent a number
- Range of numbers limited by how many binary digits available to represent the number
- Cannot represent fractions

Floating point $f \times b^e$

- Fraction (f) times base (b) raised to a power (e)
- Base b can be 2, 8, 10, 16, ...
- Can represent fractions, and very large numbers
- But, often represents numbers inexactly if there are many digits in a number (can lose the last digits)

Representing Integers

In binary, each digit (bit) represents a power of 2

A binary number:

- $b_3b_2b_1b_0$
- For example 1101 -> $b_3 = 1$, $b_2 = 1$, $b_1 = 0$, $b_0 = 1$

Each digit represents increasing power of 2

- Value = $b_3*2^3 + b_2*2^2 + b_1*2^1 + b_0*2^0$
  $= 1*8 + 1*4 + 0*2 + 1*1$
  $= 13$

With four binary digits (bits), the largest number is

- $1111 = 2^3 + 2^2 + 2^1 + 2^0 = 15$ (decimal)

Quick, in-class exercise

- What decimal number is: 0010? 0101? 0111?

Answer For The In-class Exercise

Compute the decimal value for 0010, 0101, 0111?

0010

$= b_3*2^3 + b_2*2^2 + b_1*21 + b_0*2^0$

= 0 *8 + 0*4 + 1*2 + 0*1

= 2

0101

= 0 *8 + 1*4 + 0*2 + 1*1

= 5

0111

= 0 *8 + 1*4 + 1*2 + 1*1

= 7

Representing Negative Integers

How about negative integers?

- Turns out there are several ways to do this
- Sign-and-magnitude, one's complement, two's complement, excess-n, base -2
  - http://en.wikipedia.org/wiki/Signed_number_representations
- Let's examine the easiest of these, sign-and-magnitude
  - Not in common use today, since addition in inefficient as compared to two's complement

Example:

- 0111 as a signed integer
- (sign) | $b_2*2^2 + b_1*2^1 + b_0*2^0$
- Positive + 1*4 + 2*2 + 1*1 = +7
- 1111 → -7 (the sign bit changed)

Limits of Integers

The largest number an integer can represent is limited by the number of digits available

- 8 bit signed = -128 to 127
- 16 bit signed = -32,768 to 32,767
- 32 bit signed = -2,147,483,648 to 2,147,483,647
- 64 bit signed = -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807

Representing Fractions in Floating-Point

Today, most floating point numbers follow the IEEE Standard for Floating-Point Arithmetic (IEEE 754)

- Many microprocessors implement this internally

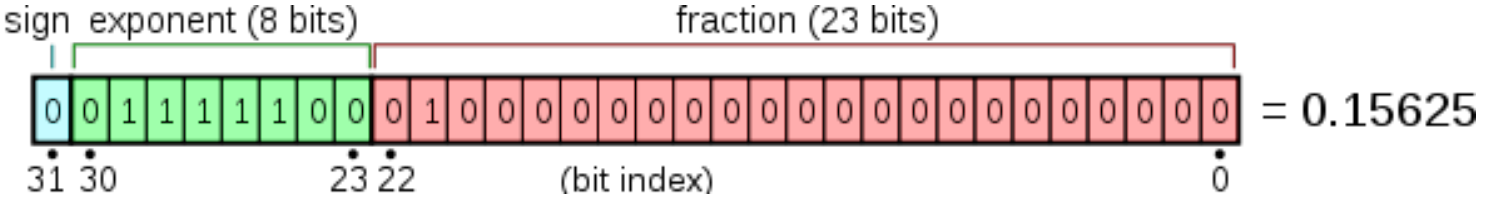
Represents FP:

- Finite numbers: $(-1)^s \times c \times b^q$
- Two infinities $-\infty$ and $+\infty$
- Not a number (NaN)
  - For situations like divide by zero, square root of negative number (imaginary numbers can't be represented)

Representing floating point

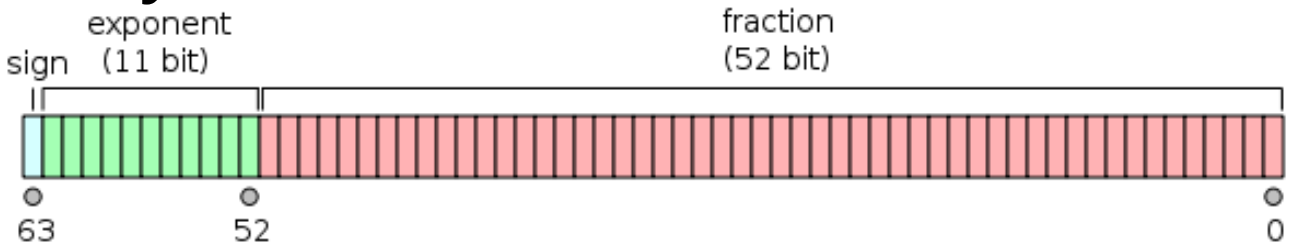
Most common forms are binary32 (single precision) and binary64 (double precision)

Binary32 (en.wikipedia.org/wiki/Binary32)



$$value = (-1)^{sign}(1 + \sum_{i=1}^{23} b_{-i}2^{-i}) \times 2^{(e-127)}$$

Binary64 (en.wikipedia.org/wiki/Binary64)



$$value = (-1)^{sign}(1 + \sum_{i=1}^{52} b_{-i}2^{-i}) \times 2^{(e-1023)}$$

Why do I Need to Know This Stuff?

Understanding the basics of how numbers are stored allows you to understand the fundamental limitations

- Can only be so big with an integer
- Can only be so precise with a float

Errors can sometimes occur due to lack of understanding

- Ariane 5 rocket exploded on June 4, 1996
- Ariane 5 reused software from the Ariane 4
- Represented acceleration in a floating point number, which is converted into a 16 bit integer
- The Ariane 5 can accelerate much faster than the Ariane 4
- Acceleration values overflowed the 16 bit integer
- Rocket went out of control

Ariane 5 Explosion

THANKS