The final

- At officially scheduled time next Tuesday at 8 AM
- Covers the material not covered on the midterm
- Not cumulative
- Multiple Choice. Bring a PINK SCANTRON, PENCILS
- Will be longer than the midterm but I don’t think for most people it will take all three hours.

- These slides select things you should know, should have learned the second half of the class.
Guest Lecture:
Robot Apocalypse
Soja Marie Morgens
Asimov's Laws: A robot.

1. Must not harm a human being through action or inaction
2. Must obey all human orders, except in conflict with Rule 1
3. Must protect its own existence, except in conflict with Rule 1 and 2
How do we prevent robots from injuring themselves or failing?

- Static Environment
- Sensors
- Coding

Often robot failure is the fault of design requirements.
How would we wish to apply rule 2?

- Gestures
- Common Speech
- Robot recognizing our patterns
- Robots learning
Learning: Which robots use learning?

Leonardo (MIT)

BigDog (Boston Dynamics)
What kinds of things could Baxter do?

Baxter
$22,000
SWARM ROBOTICS:

what might they be good for?: search and rescue.

Originals:
2003-2008
MIT/
James McLurkin

Cheap ($250)
2009-2012
Rice/
James McLurkin

Parallel:
2005-2012
Universite Libre de Bruseels/
Marco Dorigo
Guest Lecture: Prof. Sri Kurniawan
what is ergonomics?

- a. the study of people’s efficiency in their working environment
- b. the mental action or process of acquiring knowledge and understanding through thought
- c. the mental action or process of acquiring knowledge and understanding through experience
- d. the mental action or process of acquiring knowledge and understanding through the senses
The following picture was used to show:

A. Children with Autism have a very poor ability to remember.
B. Children with Autism prefer to find images on the web than to draw them themselves
C. Children with Autism have enormous difficulty even talking about the pictures they make
D. Children who normally are unable to draw well can produce creative artwork when given tools that are adapted to their limitations and strengths
E. All of the above
One fact that Prof. Kurniawan pointed out was that:

A. The best you can do with human computer interface design is to design systems that work well for males between 14 and 22

B. Kaleidoscopes are a better computational art technology

C. Black and White text on the screen is actually very bad for people with Autism

D. All of the above

E. None of the above
Prof. Kurniawan also pointed out that:

A. Everyone can be considered disabled at some point in their life
B. Assistive technologies often become useful to the mainstream population
C. HCI and AT are complementary research perspectives
D. All of the above
E. None of the above
My user group ranges from people with disabilities, the aging population, children, those w. low literacy/socio economic status and those from third world countries.
Cleft Lip and Palate

- Occurs when lip or mouth do not form correctly
- One of the most common birth defects in the world
- Corrective surgery is a very lengthy and complex process
- Speech therapy begins at 2 years; goal is to finish by kindergarten

Speech Therapy

- Speech therapy requires the patient to practice at home
- Parents are unable to motivate their children to practice
- Parents don’t really know how to assess progress

Proof-of-Concept and Prototype

- Engine and game size
- Processing time
- Recognition accuracy
- Acoustical Model Adaptation

Schematic Timeline for Treatment of Cleft Lip and Palate
Do Blind people take photos for same reasons as sighted people?

Blind Photography. Why?
- A memory of something that is emotional
  - Ex: a fun time at Mt. Fuji with your friends
- For future reference
  - Ex: a bottle of wine you liked a lot
- To share your environment and event
  - Ex: show your family your new apartment without them having to be there
- Creative expression
  - The easiest way to express visual creativity (and cheapest)
What is Prof. Kurniawan’s system for Blind Photography called?

What does it let you do?

Phodio

- Designed specifically for users who are blind (secondarily for sighted users)
- Users can capture a frame of time for personal enjoyment, as well as to share with others
  - Users take a photo with audio, time, date and location is saved with photo
  - Face detection to assist in photo aiming
- Users browse photos by listening to audio, accessing time, date, and location.
- Users can send photo to a crowdsourcing service to ask a question and receive an answer about the photo.
Guest Lecture: Computational Cinematics  
Prof. Arnav Jhala
Computer Cinematography involves:

- Question with various list elements or not.
  - lighting,
  - camera control,
  - staging,
  - writing,
  - editing,
  - viewer experience,
  - character.
A game that uses the same visual imagery as Citizen Kane is:

“The mirror causes the image to repeat infinitely. Deep focus is used to enhance the repetition, which adds to Kane's loneliness as an old man and to his isolation.”

-Aaron West, AFS
A game with the same camera continuous action shooting action as Rope is:

- MazeBall
- Panorama
- Portal

“I asked myself whether it was technically possible to film it in the same way [real-time]. The only way to achieve that, I found, would be to handle the shooting in the same continuous action, with no break in the telling of a story that begins at seven-thirty and ends at nine-fifteen.”

-Hitchcock (Interview with Truffaut)
What can Gamersourcing be used for?
Camera Games

- Afrika
- Pokemon Snap
Challenge #1: Learning Aesthetic Preferences

Which composition is better?
How would you do it automatically?
What is the rule that is used?

(a)  (b)  (c)
Data Collection

- Three methods
  - UCSC students
  - Facebook
  - Amazon’s Mechanical Turk*

- Two studies
  - Pairwise 4-AFC*
    (A is better, B is better, Both good, Both Bad)
  - Absolute (1-6: bad to good)

* Reported in this paper
Ranking scheme for collecting data on pairwise preferences that was used in the MazeBall game

(a) Likert scale (1-5)
(b) 2-AFC (alternative forced choice -- A or B)
** (c) 4-AFC (A, B, Both, Neither)
(d) None of the above

*
## Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symmetry</td>
<td>$S$</td>
<td>4</td>
</tr>
<tr>
<td>Thirds</td>
<td>$T$</td>
<td>1</td>
</tr>
<tr>
<td>Spacing</td>
<td>$Q$</td>
<td>85</td>
</tr>
<tr>
<td>Objects</td>
<td>$N$</td>
<td>1</td>
</tr>
<tr>
<td>Crops</td>
<td>$C$</td>
<td>1</td>
</tr>
<tr>
<td>Occlusions</td>
<td>$O$</td>
<td>1</td>
</tr>
<tr>
<td>Size Ratio</td>
<td>$R$</td>
<td>1</td>
</tr>
</tbody>
</table>
The Panorama game

- Aesthetic features for evaluation of photographs that are represented in the Panorama game are:
  A. Rule of Thirds
  B. Symmetry
  C. Number of objects in the scene
  D. All of the above
Player Modeling

- Blood Volume Pulse (BVP) $\rightarrow$ Heart Rate (HR)
- Skin Conductance (SC)
How to make a game more challenging/stressful?

- A. Increase number of prizes
- B. Bring the camera angle closer to the character
- C. Spread any enemies or obstacles out more evenly
- D. Use the rule of thirds
Guest Lecture: ME
Expressive Generation for Interactive Stories
Procedural Language Generation: A Key Technology

- Wide range of generation parameters
- Different methods for creating models that control the parameters
  - Dynamic Real-Time Adaptation
  - Trainable: Machine Learning Techniques
  - Individual Personalization
PERSONAGE Generator: BIG FIVE Theory

- Conscientiousness: Dutiful vs. impulsive
- Emotional stability: Calm vs. anxious
- Openness to experience: Imaginative vs. conventional
- Agreeableness: Kind vs. unfriendly
- Extraversion: Sociable, assertive vs. quiet
**1st method: Rule-Based Extraversion Generation**

- Use correlations in literature to set parameters
- Significant perceptual differences $p < .01$
- As binary classification, 90% accuracy
Scene from *Annie Hall*: Lobby of Sports Club

**ALVY**: Uh ... you-you wanna lift?
**ANNIE**: Turning and aiming her thumb over her shoulder
Oh, why-uh ... y-y-you gotta car?
**ALVY**: No, um ... I was gonna take a cab.
**ANNIE**: Laughing  Oh, no, I have a car.
**ALVY**: You have a car?
Annie smiles, hands folded in front of her
So ... Clears his throat. I don’t understand why ... if you have a car, so then-then wh-why did you say “Do you have a car?” ... like you wanted a lift?
Method

1. Collect movie scripts from IMSDb

2. Extract utterances for each character

3. Select leading roles (dialogue > 60 turns)

   - Jules’ Dialogue
   - Vincent’s Dialogue
   - Other’s Dialogue

4. Generate features reflecting linguistic behaviors

   - Jules’ LIWC results
   - Vincent’s LIWC results
   - Jules’ Tag Question Ratio
   - Vincent’s Tag Question Ratio
   - Jules’ Overall Polarity
   - Vincent’s Overall Polarity
   - Jules’ other features
   - Vincent’s other features

The web’s largest movie script resource!
Has anyone ever shown that it matters?

- People make attributions beyond social level: task competence
- Personality matching in a robotic exercise coach increased the time that stroke victims spent on their medically recommended exercises (Tapus & Mataric 2008)
- Tutoring oriented to the student’s ‘face needs’ improved learning in training and tutoring (Porayska-Pomsta & Mellish 2004; Wang et al., 2004)
Making your own web page. Basics of HTML
What are you supposed to learn?

- HTML let’s you programmatically indicate how a particular content should be displayed.
- It can be served up by any HTTP server anywhere in the world.
- Typically uses UTF-8 encoding to guarantee being able to be shown.
DATA and SOFTWARE DOWNLOADS.

PROFESSIONAL EXPERIENCE

Professor of Computer Science, Natural Language and Dialogue Systems Lab, University of California, Santa Cruz, 2009 to present

Professor of Computer Science, Head of Cognitive Systems Group, University of Sheffield, Sheffield, England, 2003 to 2009

Principal Research Staff Member, ATT Labs - Research, Florham Park, N.J., Speech Processing Software and Technology Research, 1996 to 2003

Research Scientist, Mitsubishi Electric Research Laboratories, Cambridge, Ma., Interactive Learning and Entertainment, 1993 to 1996

Consultant, Hewlett Packard Laboratories, Bristol, England, on dialogue systems, speech technology, and personal information systems: 1989-1993

Researcher, Dialogue Modeling Department, Electrotechnical Laboratory, Tsukuba City, Japan: Summer 1991


EDUCATION


B.A. Computer and Information Science, With Honors, University of California Santa Cruz, 1984.

http://users.soe.ucsc.edu/~maw/
Professor Marilyn Walker
Department of Computer Science
University of California Santa Cruz
1156 N. High, SOE-3
Santa Cruz, Ca. 95064
Phone: 831 429 1058

<a href="mailto:maw@soe.ucsc.edu">maw@soe.ucsc.edu</a>

DATA and SOFTWARE DOWNLOADS

PROFESSIONAL EXPERIENCE

Professor of Computer Science, Natural Language and Dialogue Systems Lab, University of California, Santa Cruz, 2009 to present
Professor of Computer Science, Head of Cognitive Systems Group, University of Sheffield, Sheffield, England, 2003 to 2009
Principal Research Staff Member, ATT Labs - Research, Florham Park, N.J., Speech Processing Software and Technology Research, 1996 to 2003
Research Scientist, Mitsubishi Electric Research Laboratories, Cambridge, Ma., Interactive Learning and Entertainment, 1993 to 1996
Consultant, Hewlett Packard Laboratories, Bristol, England, on dialogue systems, speech technology, and personal information systems: 1989-1993
Researcher, Dialogue Modeling Department, Electrotechnical Laboratory, Tsukuba City, Japan: Summer 1991

EDUCATION


Informational Redundancy and
HTML and the Web

- The Web uses http:// protocol
- Its asking for a Web page, which usually means a page expressed in hyper-text markup language, or HTML
  - *Hyper-text* refers to text containing LINKS that allow you to leave the linear stream of text, see something else, and return to the place you left
  - *Markup language* is a notation to describe how a published document is supposed to look: what kinds of fonts, text color, headings, images, etc.
Basics of HTML #1

- Rule 1: Content is given directly; anything that is not content is given inside of tags
- Rule 2: Tags made of < and > and used this way:

  <p style="color:red">This is paragraph.</p>

  Start Tag
  Content
  End Tag

  It produces: This is paragraph.
- Rule 3: Tags must be paired or “self terminated”
Basics of HTML  #2

- **Rule 4:** An HTML file has this structure:

  ```html
  <html>
  <head><title>Name of Page</title></head>
  <body>
  Actual HTML page description goes here
  </body>
  </html>
  ```

- **Rule 5:** Tags must be properly nested

- **Rule 6:** White space is mostly ignored

- **Rule 7:** Attributes (`style="color: red"`) preceded by space, name not quoted, value quoted
Basics of HTML #3

- To put in an image (.gif, .jpg, .png), use 1 tag
  
  `<img src="cooking-ewan-isabel.jpg" alt="Kids Cooking"/>

- To put in a link, use 2 tags
  
  `<a href=http://users.soe.ucsc.edu/~maw>Prof. Walker’s</a>`
Example: myfirst.html

<html>
<head>
<title>Fun in the Kitchen</title>
</head>
<body>
<h1>Making a Bake</h1>
<img src="cooking-ewan-isabel.jpg" alt="Kids Cooking" width="404" height="328" />
<p>When do we get to eat it?!</p>
</body>
</html>
In Processing: File Export

- Makes an index.html file. You browse it!
Important Processing Concepts
Processing Concepts you NEED TO KNOW!

- Variables and Declarations
- Assignments
- Expressions
- Tests or If-statements

- NEWEST STUFF.
- Repetition (looping) or For-statements
- Functions and Functional Abstraction
For loops (Repetition)

- Repeating commands is powerful:
  - Lightbot 2.0 used recursion, a function calling itself
  - Symbolic Lightbot prefixed a number, 2:Step
- Processing uses a `for` loop:

```java
void setup() {
    size(500,200);
    background(0);
    noStroke();
    smooth();
    fill(255);
    for (int i=0; i < 16; i++) {
        ellipse(100+25*i, 100, 15, 15);
    }
}
```
Or how about a bullseye?

Note the *loop variable* must be declared ... but could do it in loop itself like we did for pacman pills:

```cpp
for (int i = 0; ...
```
Functional Abstraction Reduces Complexity

Layering: Building Functions out of Functions
Example: Abstraction in Everyday Life

- Get Dressed
  - Dress Bottom Half
    - Put on Sox
  - Dress Top Half
    - Put on Pants
    - Put on Shirt
Abstraction

- Abstraction is a way to understand and solve problems
- As a general definition, abstraction eliminates details to focus on essential properties
- Functional abstraction gives a name to a series of operations that perform a coherent (and to us meaningful) activity; the name is the instruction, the series of operations are the actions to implement it
Functions In Processing: Result

- Functions that do something, but do not return a value, have **void** as their *return type*
- Functions that return a value must say its type

```java
void draw_a_box (int x_pos, int y_pos) {
    rect(x_pos, y_pos, 20, 20);
}

color pink () {
    return color(255, 200, 200);
}
```
Homework 10: Functions.

- Make sure you can reproduce what you did during final (e.g. be sure to understand the solution code)
- `void cell(int x, int y, int s, color tinto)`
- `void triple(int x, int y,`  
  
  Also functions for
  - `block( )`
  - `row( )`
  - `cellarray( )`

**Sudoku GUI**
The Sudoku board is a 9×9 array of cells. Alternating 3×3 cells are colored gray and strong lines separate all of the 3×3 cells, as follows. (More information at http://www.websudoku.com/.)
Functional Abstraction: Layers of Functions

three_digit

sel

digit one two three four five six seven eight nine zero

leftupper leftlower tophoriz midhoriz bothoriz rightupper rightlower

hexa rexa

triangle, rect, triangle
Using Functions

- Once defined, functions can be called repeatedly ... it’s the point of writing them!

```c
void setup( ) {
    size(110, 100);
    background(0);
    noStroke();
}

void draw( ) {
    fill(255);
    hexa(20, 20);
    hexa(50, 20);
    hexa(80, 20);
}

void hexa(float xbase, float ybase) {
    rect(xbase, ybase+10, 20, 40);
    triangle(xbase, ybase+10, xbase+20, ybase+10, xbase+10, ybase);
    triangle(xbase, ybase+50, xbase+20, ybase+50, xbase+10, ybase+60);
}
```
Arguments Become Parameters

- Notice that if the DEFINITION has \( n \) parameters, the CALL needs \( n \) arguments
- The parameters and arguments correspond

```c
void draw( ) {
    fill(255);
    hexa(20, 40);
    hexa(50, 40);
    hexa(80, 40);
}

void hexa(float xbase, float ybase) {
    rect(xbase, ybase+10, 20, 40);
    triangle(xbase, ybase+10, xbase+20, ybase+10, xbase+10, ybase);
    triangle(xbase, ybase+50, xbase+20, ybase+50, xbase+10, ybase+60);
}
```

Inside of the function, the parameter, e.g. xbase, is declared and initialized to the corresponding argument, e.g. 80. Then, the definition uses it, e.g. `rect(80, 40+10, 20, 40)`
Parameters

- Parameters are automatically declared (and initialized) on a call, and remain in existence as long as the function remains unfinished.

- When the function ends, the parameters vanish, only to be recreated on the next call.

- It is wise to choose parameter names that help you remember exactly what they mean:
  - colorFlag (Chao’s code),
  - dir (for direction)
Example: Clock Timer. In sample codes.

- Draw digital timer elements
- Assemble elements into digits
- Light digit segments to create numbers
- Select number based on a digit
Functional Abstraction: Layers of Functions

three_digit

sel
digit one two three four five six seven eight nine zero

leftupper leftlower tophoriz midhoriz bothoriz rightupper rightlower

hexa rexa

triangle, rect, triangle
Using Functions

- Once defined, functions can be called repeatedly ... it’s the point of writing them!

```cpp
void setup() {
  size(110, 100);
  background(0);
  noStroke();
}

void draw() {
  fill(255);
  hexa(20, 20);
  hexa(50, 20);
  hexa(80, 20);
}

void hexa(float xbase, float ybase) {
  rect(xbase, ybase+10, 20, 40);
  triangle(xbase, ybase+10, xbase+20, ybase+10, xbase+10, ybase);
  triangle(xbase, ybase+50, xbase+20, ybase+50, xbase+10, ybase+60);
}
```
Clicker: I have hexa() and rexa(), now what?

A. I can make fun little shapes
B. It obvious how I can make digits given these two shapes
C. I have to define at least two more functions before I can make digits
D. I think I can make digits but I just can’t see how.
Use hexa, rexa to make a Digit

- hexa_rexa_makeadigit.pde

```pde
void draw() {
    fill(255);
    digit(50, 20);
    digit(140, 20);
}

void hexa(float xbase, float ybase) {
    rect(xbase, ybase+10, 20, 40);
    triangle(xbase, ybase+10, xbase+20, ybase+10, xbase+10, ybase);
    triangle(xbase, ybase+50, xbase+20, ybase+50, xbase+10, ybase+60);
}

void rexa(float xbase, float ybase) {
    triangle(xbase, ybase+10, xbase+10, ybase, xbase+10, ybase+20);
    rect(xbase+10, ybase, 40, 20);
    triangle(xbase+50, ybase, xbase+50, ybase+20, xbase+60, ybase+10);
}

void digit(float xbase, float ybase) {
    hexa(xbase, ybase+10); //left upper
    hexa(xbase, ybase+70); //left lower
    rexa(xbase+10, ybase); //mid horizontal
    rexa(xbase+10, ybase+60); //top horizontal
    rexa(xbase+10, ybase+120); //bot horizontal
    hexa(xbase+60, ybase+10); //right upper
    hexa(xbase+60, ybase+70); //right lower
}
```
Clicker: What does digit help you do?

A. You can re-use it over and over again
B. It tells you what the coordinates are for the various bits of the digit
C. If you pass the right arguments for xbase and ybase to digit you can make every digit
D. A & B
E. None of the above

```c
void digit(float xbase, float ybase) {
    hexa(xbase, ybase+10);    // left upper
    hexa(xbase, ybase+70);    // left lower
    rexa(xbase+10, ybase);    // mid horizontal
    rexa(xbase+10, ybase+60); // top horizontal
    rexa(xbase+10, ybase+120); // bot horizontal
    hexa(xbase+60, ybase+10); // right upper
    hexa(xbase+60, ybase+70); // right lower
}
```
Let There Be Light (and Dark)

- Define the illumination of the digit

  `hexa_rexa_makeadigit_addlight.pde`

  - Must declare two color variables, initialize to proper colors, use them in `fill`, and then check them

```pde
void setup( ) {
    size(250, 180);
    background(0);
    stroke(0);
}

void draw( ) {
    lite = color(255, 185, 0);
    dark = color(64, 48, 0);
    fill(dark);
    digit(50, 20);
    fill(lite);
    digit(140, 20);
}
```
Clicker: What does the lighted digit help you do?

A. You can re-use it over and over again
B. It tells you what the coordinates are for the various bits of the digit
C. If you pass the right arguments for xbase and ybase to digit you can make every digit
D. A & B
E. None of the above

void digit(float xbase, float ybase) {
    hexa(xbase, ybase+10);    //left upper
    hexa(xbase, ybase+70);    //left lower
    rexa(xbase+10, ybase);    //mid horizontal
    rexa(xbase+10, ybase+60); //top horizontal
    rexa(xbase+10, ybase+120); //bot horizontal
    hexa(xbase+60, ybase+10); //right upper
    hexa(xbase+60, ybase+70); //right lower
}
What is time doing?

A. Nothing. It has a bug in it.
B. It is counting to infinity.
C. It is counting to 999 then starting at zero again
D. It counts up to 1000 then starts at 1.

```cpp
void draw() {
    lite = color(255,185,0);
    dark = color(64, 48, 0);
    if (stop == 1) {
        fill(dark);
        digit(50,90);
        digit(140, 90);
        digit(260, 90);
        fill(lite);
        rect(235, 155, 15, 15);
        time=(time+1)%1000;
        sel(time%10, 260, 90);
        sel((time/10)%10, 140, 90);
        sel(time/100, 50, 90);
    }
}
```
Clicker: this code for writing the digit 4 is …

```c
void four (float xbase, float ybase) {
    hexa(xbase, ybase);        //left upper
    hexa(xbase+60, ybase);     //right upper
    rexa(xbase+10, ybase+10);  //mid horizontal
    hexa(xbase+60, ybase+60);  //right lower
}
```

A. Perfect just as it is
B. Has a bug for right upper
C. Has a bug for mid horizontal
D. Actually makes a five
Count_in_lights_three_level

```c
void digit(float xbase, float ybase) {
    //hexa(xbase, ybase+10);       //left upper
    //hexa(xbase, ybase+70);       //left lower
    leftupper (xbase,ybase);
    leftlower (xbase,ybase);
    //rexa(xbase+10, ybase);       //top horizontal
    tophoriz (xbase,ybase);       //top horizontal
    rexa(xbase+10, ybase+60);     //mid horizontal
    rexa(xbase+10, ybase+120);    //bot horizontal
    hexa(xbase+60, ybase+10);     //right upper
    hexa(xbase+60, ybase+70);     //right lower
}

void leftupper (float xbase, float ybase) {
    hexa(xbase, ybase+10);        //left upper
}

void leftlower (float xbase, float ybase) {
    hexa(xbase, ybase+70);        //left lower
```
Clicker: why would you want `Count_in_lights_three_level`?

A. I wouldn’t want it, it just makes the code even more confusing

B. Adding numbers to `ybase` and `xbase` for rexa and hexa can cause silly errors.

C. It makes the code more self documenting.

D. B & C

```c
void digit(float xbase, float ybase) {
    //hexa(xbase, ybase+10);    //left upper
    //hexa(xbase, ybase+70);     //left lower
    leftupper (xbase,ybase);
    leftlower (xbase,ybase);
    //rexa(xbase+10, ybase);    //top horizontal
tophoriz (xbase,ybase);     //top horizontal
    rexa(xbase+10, ybase+60);  //mid horizontal
    rexa(xbase+10, ybase+120);  //bot horizontal
    hexa(xbase+60, ybase+10);  //right upper
    hexa(xbase+60, ybase+70);  //right lower
}

void leftupper (float xbase, float ybase) {
    hexa(xbase, ybase+10);  //left upper
}

void leftlower (float xbase, float ybase) {
    hexa(xbase, ybase+70);  //left lower
}
Do function names matter?
Web Search

Needles in the Haystack: Google and Other Brokers in the Bits Bazaar

Blown To Bits Chapter 4
Important questions: What you should learn

- How can a search engine respond so fast?
- Does it find every relevant link?
- How does a search engine decide what gets listed first?
- If you try another search engine will you get the same result? If so, which is right? Which is better? Which is more authoritative?
- Are sponsored links better than “organic” links?
- Is the advertising necessary?
- What is the role of government? What should it be?
1. Gather information.
2. Keep copies.
3. Build an index.
4. Understand the query.
5. Determine the relevance of each possible result to the query.
6. Determine the ranking of the relevant results.
7. Present the results.
Search Engines

No one controls what’s published on the WWW ... it is totally **decentralized**

To find out, *search engines crawl* Web

- Two parts
  - *Crawler* visits Web pages building an *index* of the content (stored in a database)
  - *Query processor* checks user requests against the index, reports on known pages [You use this!]

*Only a fraction of the Web’s content is crawled*
How many hosts are out there?
1. Gather Information

- Spiders or web crawlers wander the web building indices
- Estimates range from .02% to 3% of information is indexed
- How often does a page get visited?
  - some frequently (daily whitehouse.gov), others rarely
  - Crawler keeps track of which pages change frequently
2. Keep Copies

- Spider downloads the page as part of the “visit” in order to create the index.
- Search engine may “cache” the copy.
- Is this legal? What about copyright?
- But wait, browsing requires copying as well.

“(AFP) – Sep 15, 2011

NEW YORK — Google and publishers told a US judge Thursday they are close to settling a lawsuit over the Internet giant's controversial book-scanning project…”
Clickers

- Google responds efficiently to queries by going out and searching the web in real time.

A. TRUE.
B. FALSE
Which steps happen in the background?

A. Gather information & Keep copies.
B. Gather information & Keep copies & Build an index.
C. Understand the query.
D. Determine the relevance of each possible result to the query & Determine the ranking of the relevant results.
Building an Index
3. Build an Index

- list of terms and for each term a list of where it appeared
- more than just the terms
  - terms in bigger font might be more important
  - terms in the title might be more important
- must be very fast to lookup
- could be millions of entries (not just words, but names, special numbers, etc.) requiring Gigabytes of memory
- must fit in the computers memory (see next slide)
Google indexes what percent of the pages in the world?

A. 100% It indexes everything.
B. About 50% on average but some days it’s 100% and some days it’s 30%
C. Less than 5%
D. Best estimates put it at 30% because of all the bots that keep crawlers out.
How does it find the pages to index?

A. Every website has to register with Google to get indexed
B. Every website has to pay Google to get indexed
C. Google knows which pages to index because of your digital footprint
D. Google has a list of ‘trusted pages’ and it just follows the links from them
How often does a page get visited?

A. Every page that gets indexed is visited every day.
B. It depends on the page, pages like whitehouse.gov get visited daily others rarely
C. Google decides by keeping track of how often pages change.
D. B & C
E. None of the above.
Clickers: What goes into the short index?

A. Every word on the page
B. The words that other sites use when they point to (link to) this page.
C. Only the keywords on the page.
D. Google has a list of special keywords that all pages get indexed by.
Google Short Index & Long Index

- Known as the short barrels and the long barrels..
- Short index:
  - store the words in link texts that point to a page (inbound links!!)
  - the words in a page’s title, and one or two other special things.
- The link text words are attributed to the target page, and not to the page that the link is on.
  - In other words, if my page links to your page, using the link text “Miami hotels”, then the words “Miami” and “hotels” are stored in the short index as though they appeared in your page, but they belong to my page. If 100 pages link to your page, using those same words as link text, then your page will have a lot of entries in the short index for those particular words.
- The long index is used to store all the other words on a page – its actual content.
Understanding your query: Boolean queries
Clickers

- Google responds efficiently to queries by going out and searching the web in real time.

A. TRUE.
B. FALSE
How does the index get used at search time by default? (without advanced search)

A. Every website registers with Google exactly which terms to index by and which combinations

B. Google uses “boolean” combinations. The index is made of single words. Google ANDS them together and finds which webpages (URLs) are in the intersection of all the terms

C. Google indexes the pages individually for each person using your digital footprint

D. All of the above.
Make A Query: [http://www.google.ca/advanced_search](http://www.google.ca/advanced_search)

- When Google gets the query
- It “ands” the two lists together, finding URLs that are on both lists
- It counts them up, records time, shows 10 hits
Processing a query: short and long indices

- First try to get enough results from the short index.
- If you can’t get enough results, then use the long index to add to what they have.
  - It means that, if they can get enough results from the short index – that’s the index that contains words in link texts and page titles – then they don’t even look in the long index where the actual contents of pages are stored. Page content isn’t even considered if they can get enough results from the link texts and titles index – the short index.

- Thus: link texts are very powerful for Google rankings.
  - Much more powerful than page titles, because a page can have the words from only one title in the short index, but it can have the words from a great many link texts in there.
  - Page titles and meta descriptions were the second most powerful ranking factors, because they are stored in the short index.
QA & NLP: Beyond Boolean Queries

Search Engine words are independent

- Words don’t have to occur together
- Use Boolean queries and quotes
- Logical Operators: AND, OR, NOT
  - monet AND water AND lilies
  - “van gogh” OR gauguin
  - vermeer AND girl AND NOT pearl
How does it rank the pages it finds?

A. It uses what is called a “page rank” algorithm, that uses many different factors
B. It depends only on who is willing to pay the most.
C. The government tells it how to rank pages.
D. None of the above.
The “crown jewels” of search engines lie in their page rank algorithms.

Factors include:

- keywords in heading or titles
- keyword only in the body text
- site is “trustworthy”
- links on this page are to relevant pages
- links to this page are relevant
- age of the page
- quality of the text (e.g. absence of misspellings)
The information that Page Rank uses includes:

- Keywords in heading or titles and keywords in the body text
- Information about whether the site is “trustworthy”
- Whether the **LINKS ON THIS PAGE ARE TO** relevant pages
- Whether the **LINKS TO THIS PAGE** are relevant
- Age of the page
- Quality of the text (e.g. absence of misspellings)

It uses what is called a “page rank” algorithm, that uses many different factors.
Its all free?? : Well no. Who Pays for What?

- Users could pay a subscription fee (early AOL and CompuServe)
- Web sites could pay for being indexed.
- The government could pay (taxes?).
- Advertisers could pay.

And it matters who pays cause it affects how it works
What makes search attractive to advertisers?

A. Advertisements can be targeted more precisely using your digital footprint.
B. Advertisers only pay when you click on their ad.
C. Millions of people use search every day.
D. All of the above.
Advertiser only pays when you click!

Costs

AdWords differs from traditional forms of advertising. It doesn’t come with price lists or rate cards. Instead, our standard pricing model relies on a cost-per-click (CPC) based auction:

How much you invest is up to you

You’ll start by choosing a daily budget you’re comfortable with, and then change it at any time. There’s no minimum spend and you’re not locked into a contract.

Only pay for results

With ads on Google you’re not charged when your ad is displayed, but only when someone clicks on your ad to go to your website. In other words, you only pay when your advertising works.

You’re in control of your cost-per-click

We set cost-per-click prices via an automated auction system. As an advertiser, you get to specify the maximum amount (max CPC bid) you’re willing to pay for each prospective customer that visits your website by clicking on your ad. This bid then determines how high up on the search results page your ad appears, which eventually determines how often people notice and click on your ad.

Relevance pays off

Your bid isn’t the only factor that counts. AdWords rewards more relevant ads with a higher Quality Score and
Clickers

- The federal government uses your tax dollars to guarantee that Google and other search engine providers like Microsoft (Bing) return the objectively best results of your query.

A. TRUE
B. FALSE
Clickers

- Web search is free and democratic. Every web page has an equal chance of being indexed and coming out the top of the list.

A. TRUE
B. FALSE
Search engine possible funding models:

A. Users could pay a subscription fee
B. Web sites could pay for being indexed.
C. The government could pay using taxes the same way they pay for roads or police.
D. Advertisers could pay for having their ads featured in the side bar.
E. All of the above are possible.
Search engine current funding model:

A. Users pay a subscription fee
B. Web sites pay for being indexed.
C. The government pays using taxes the same way they pay for roads or police.
D. Advertisers pay for having their ads featured in the side bar.
E. B&D
Artificial Intelligence as Question Answering. IBM Watson on Jeopardy
The Turing Test

- Turing in 1950 published a philosophical paper designed to stop people arguing about whether or not machines could think.
- He proposed that the question be replaced with a test, which is what is now called the Turing Test.

Vol. lix. No. 236.) [October, 1950]

MIND
A QUARTERLY REVIEW OF
PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing
What is the underlying technology called?
Question Answering (QA)

This is a Natural Language Processing Technology
I am going to skip over this today, since it is more recent to make sure I have time to cover the stuff we have not reviewed yet. But be sure to look at the clicker questions in here.
Question Answering: A type of NLP
Roots of Question Answering

- Information Retrieval (IR)
- Information Extraction (IE)

Motivated by the observation that Information Retrieval does not answer your question, it provides a ranked list of documents that *might* contain the answer to your question.
Contrast IR/Search with Asking an Expert

Decision Maker
- Has Question
- Distills to 2-3 Keywords
- Reads Documents, Finds Answers
- Considers Answer & Evidence

Search Engine
- Finds Documents containing Keywords
- Delivers Documents based on

Expert
- Understands Question
- Produces Possible Answers & Evidence
- Analyzes Evidence, Computes Confidence
- Delivers Response, Evidence & Confidence

Enabling Technologies – The Time Was Right

**Natural Knowledge**
- Large volumes of natural language electronic text (e.g., news, wikis, reference, web, etc.)
- Encodes knowledge and greater linguistic contexts to better resolve intended meaning

**Semi-Structured Knowledge**
- Large volumes of Thesauri, Dictionaries, Folksonomies, Linked Data, and the Semantic Web
- Rapid, community-based construction
- Across many domains – Specialized and General

**NLP (Text Analysis)**
- Entity and Relation Detection, Syntactic & Semantic Parsing
- Statistical NLP - Broader coverage, lower cost Information Extraction
- Statistical Paraphrasing: Learn ways to express same meaning

**Compute Power**
- Massive parallel compute power
- 1000s of compute cores working simultaneously
- TBs of globally addressable main memory
DeepQA: The Technology Behind Watson

Massively Parallel Probabilistic Evidence-Based Architecture

Generates and scores many hypotheses using a combination of 100's of Natural Language Processing, Information Retrieval, Machine Learning and Reasoning Algorithms. These gather, evaluate, weigh and balance different types of evidence to deliver the answer with the best support it can find.
What technologies made Watson possible?

A. The Web
B. Compute Power
C. Question Answering
D. Natural Language Processing
E. All of the above.
How long have people been working on it?

- TREC = Text REtrieval Conferences
  - Series of annual evaluations, started in 1992
  - Organized into “tracks”
- Test collections are formed by “pooling”
  - Gather results from all participants
  - Corpus/topics/judgments can be reused
- TREC has had a QA Track since 1999.
Types of Question Answering

- **Factoid**
  - Who discovered oxygen?
  - When did Hawaii become a state?
  - Where is Ayers Rock?
  - What team won the World Series in 1992?
- **List**
  - What countries export oil?
  - Name U.S. cities that have a “Shubert” theater.
- **Definition**
  - Who is Aaron Copland?
  - What is a quasar?
Central Idea of Factoid QA

- Determine the semantic type of the expected answer
  
  “Who won the Nobel Peace Prize in 1991?” is looking for a PERSON

- Retrieve documents that have keywords from the question

  Retrieve documents that have the keywords “won”, “Nobel Peace Prize”, and “1991”

- Look for named-entities of the proper type near keywords

  Look for a PERSON near the keywords “won”, “Nobel Peace Prize”, and “1991”
Answer Type Hierarchy

Top

REASON TIME PRODUCT ORGANIZATION MANNER NATIONALITY ALPHABET MAMMAL REPTILE AUTHOR LANDMARK

DATE MONEY LOCATION LANGUAGE PERSON QUOTATION NUMERICAL GAME DOG BREED DEFINITION VALUE

UNIVERSITY COUNTRY CONTINENT

DEGREE RATE PERCENTAGE AMOUNT SPEED

CITY PROVINCE OTHER LOC

DIMENSION DURATION COUNT TEMPERATURE

scientist biologist botanist

performer actor dancer musician

European Italian Romanian

Swede guardian defender

bodyguard escort custodian

girth philosopher yogi

easternist pragmatist
Factoid QA

- The central idea of Factoid questions is to determine the semantic type of the expected answer. Some of the possible semantic types of expected answers are:

A. Persons and Places
B. Distances, Heights, Lengths and other numerical values
C. Universities, Businesses, Organizations
D. Types of Animals, such as Reptile or Mammal
E. All of the above
The types of questions in question answering research are:

A. Definition Questions: e.g. Who is Jared Borgetti?
B. Factoid Questions: e.g. How many feet above sea level is the UCSC Campus?
C. List Questions: e.g. Name the players on Brazil’s World Cup Winning Team in 2002.
D. None of the above
E. All of the above
But many foreign investors remain sceptical, and western governments are withholding aid because of the Slorc's dismal human rights record and the continued detention of Ms Aung San Suu Kyi, the opposition leader who won the Nobel Peace Prize in 1991.

The military junta took power in 1988 as pro-democracy demonstrations were sweeping the country. It held elections in 1990, but has ignored their result. It has kept the 1991 Nobel peace prize winner, Aung San Suu Kyi - leader of the opposition party which won a landslide victory in the poll - under house arrest since July 1989.

The regime, which is also engaged in a battle with insurgents near its eastern border with Thailand, ignored a 1990 election victory by an opposition party and is detaining its leader, Ms Aung San Suu Kyi, who was awarded the 1991 Nobel Peace Prize. According to the British Red Cross, 5,000 or more refugees, mainly the elderly and women and children, are crossing into Bangladesh each day.
First finding: Cannot Anticipate Answer Types!!

Broad Domain

We do NOT attempt to anticipate all questions and build specialized databases.

In a random sample of 20,000 questions we found 2,500 distinct types*. The most frequent occurring <3% of the time. The distribution has a very long tail.

And for each these types 1000’s of different things may be asked.

Even going for the head of the tail will barely make a dent

*13% are non-distinct (e.g., it, this, these or NA)

Our Focus is on reusable NLP technology for analyzing vast volumes of as-is text. Structured sources (DBs and KBs) provide background knowledge for interpreting the text.
This graph from Prof. Walker’s lecture on IBM’s Watson was used to demonstrate that:

A. It is always possible to determine the semantic type of the expected answer
B. Anticipating the question types and building specialized databases would not let Watson win at Jeopardy
C. Most questions have to do with Distances, Heights, Lengths and other numerical values
D. Online encyclopedias have most answers already pre-computed in an easy to extract format
E. None of the above
Interesting tradeoff: Knowledge, Precision, Open Domain

Structured KB approach delivers high confidence if questions can be precisely mapped to existing & reliable sources. Turns out to be rarely the case and confidence and accuracy drop off quickly.

Basic Text Search approach never delivers high confidences but quickly reaches and maintains a peak accuracy at about 30%.

Must combine deep and shallow semantic analysis over structured & unstructured content to drive up precision, recall and confidence.
Wikipedia: What knowledge can we get from Wikipedia?

Using WordNet: Online Thesaurus.

- [http://wordnetweb.princeton.edu/](http://wordnetweb.princeton.edu/)
- What is the service ceiling of a U-2?
- Can access it FROM a program (not just this interface).
What kind of information does Wordnet have?
Using WordNet: Online Thesaurus.

- What is the service ceiling of a U-2
- Can access it as a program.

http://wordnetweb.princeton.edu/

**WordNet Search - 3.1**
- WordNet home page - Glossary - Help

Word to search for: ceiling

Display Options: (Select option to change) Search WordNet

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: (gloss) "an example sentence"

**Noun**
- S: (n) ceiling (the overhead upper surface of a covered space) "he hated painting the ceiling"
- S: (n) ceiling ((meteorology) altitude of the lowest layer of clouds)
- S: (n) ceiling, roof, cap (an upper limit on what is allowed) "he put on the number of women who worked for him"; "there was a roof on salaries"; "they established a cap for prices"
- S: (n) ceiling (maximum altitude at which a plane can fly (under specified conditions))
  - direct hyponym / full hyponym
  - direct hypernym / inherited hypernym / sister term
    - S: (n) altitude, height (elevation especially above sea level or above the earth's surface) "the altitude gave her a headache"
      - S: (n) level (height above ground) "the water reached ankle level"; "the pictures were at the same level"
    - S: (n) ceiling (meteorology) altitude of the lowest layer of clouds
    - S: (n) ceiling (maximum altitude at which a plane can fly (under specified conditions))
Does Wordnet know about Chocolate Cake?

WordNet Search - 3.1
- WordNet home page - Glossary - Help

Word to search for: chocolate cake

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: (gloss) "an example sentence"

Noun

- S: (n) chocolate cake (cake containing chocolate)
  - direct hyponym / full hyponym
    - S: (n) devil's food, devil's food cake (very dark chocolate cake)
  - direct hypernym / inherited hypernym / sister term
Chocolate Decadence Cake?: The Web knows all!

Chocolate Decadence Cake Recipe - Allrecipes.com
allrecipes.com/recipe/chocolate-decadence-cake-
65 reviews - 425 cal
This is the richest chocolate cake ever! Any questions?
Recipes like this - Recipe Reviews - Photos of this recipe

Chocolate Decadence Recipe - Allrecipes.com
allrecipes.com/recipe/chocolate-decadence/!
★★★★★ 10 reviews - 4 hrs 45 mins - 250 cal
Six dainty cups of deeply rich chocolate are gently baked and served cooled to a small
grateful crowd.

Chocolate Decadence Cake III Recipe - Allrecipes.com
allrecipes.com/recipe/chocolate-decadence-cake-iii/
★★★★☆ 6 reviews - 1 hr - 324 cal
This is a very rich cake, a small piece goes a long way. The best way to describe it is
that it's a cross between a moist brownie and fudge. Be sure to use good...
Structured knowledge vs. text search (clicker)

What does this graph show?

A. Structured knowledge sources can answer most of the questions with high confidence
B. Text search by itself can never answer more than 30% of the questions
C. Structured knowledge by itself can answer at least 50% of the questions with 50% confidence
D. Watson must combine deep (natural language understanding) and shallow (words and indices) semantic analysis to win
A Few Guiding Principles

- Specific large hand-crafted models won’t cut it
  - Too slow, too narrow, too brittle, too biased
  - Need to acquire and analyze information from as-is knowledge sources

- Intelligence from many diverse methods
  - Many diverse algorithms must be combined: No single one is expected to solve the whole problem. Each addressing different weaknesses.
  - Relative impact of many overlapping methods must be learned

- Massive Parallelism is a Key Enabler
  - Pursue many competing independent hypotheses over large data
  - Efficiency will demand simultaneous threads of evidence evaluation
This is not easy!

Example Question

IN 1698, THIS COMET DISCOVERER TOOK A SHIP CALLED THE PARAMOUR PINK ON THE FIRST PURELY SCIENTIFIC SEA VOYAGE

Keywords: 1698, comet, paramour, pink, ...
AnswerType(comet discoverer)
Date(1698)
Took(discoverer, ship)
Called(ship, Paramour Pink)

Related Content (Structured & Unstructured)

Primary Search

Candidate Answer Generation

Term Overlap
Classification
Relations
Temporal

Evidence Retrieval

Evidence Scoring

Isaac Newton
Wilhelm Tempel
HMS Paramour
Christiaan Huygens
Halley's Comet
Edmond Halley
Pink Panther
Peter Sellers

[0.58 0.5 -1.3 ... 0.97]
[0.71 1 13.4 ... 0.60]
[0.42 0 2.0 ... 0.90]
[0.84 0.5 10.6 ... 0.88]
[0.33 0 6.3 ... 0.83]
[0.21 1 11.1 ... 0.92]
[0.91 0 -8.2 ... 0.31]
[0.91 0 -1.7 ... -0.20]

1) Edmond Halley (0.85)
2) Christiaan Huygens (0.20)
3) Peter Sellers (0.05)

Merging & Ranking

© 2011 IBM Corporation
In cell division, mitosis splits the nucleus & cytokinesis splits this **liquid cushioning** the nucleus.

- Many candidate answers (CAs) are generated from many different searches.
- Each possibility is evaluated according to different dimensions of evidence.
- **Just One** piece of evidence is if the CA is of the right type. In this case a “liquid”.

\[
\begin{align*}
\text{Is(“Cytoplasm”, “liquid”) } &= 0.2 \\
\text{Is(“organelle”, “liquid”) } &= 0.1 \\
\text{Is(“vacuole”, “liquid”) } &= 0.2 \\
\text{Is(“plasma”, “liquid”) } &= 0.7
\end{align*}
\]

“Cytoplasm is a **fluid surrounding** the nucleus…”

Wordnet \(\rightarrow\) Is_a(Fluid, Liquid) \(\rightarrow\) ?

Learned \(\rightarrow\) Is_a(Fluid, Liquid) \(\rightarrow\) yes.
Potential Business Applications

**Healthcare / Life Sciences:** Diagnostic Assistance, Evidenced-Based, Collaborative Medicine

**Tech Support:** Help-desk, Contact Centers

**Enterprise Knowledge Management and Business Intelligence**

**Government:** Improved Information Sharing and Security
Creativity in the kitchen?

- [http://www.psfk.com/2013/03/ibm-watson-top-chef.html](http://www.psfk.com/2013/03/ibm-watson-top-chef.html)

IBM's Watson is an artificial intelligence computer system that was developed to solve complex problems and is capable of answering questions posed in "human speak." Most recently, the supercomputer collaborated with chef instructor James Briscolone to create a "Spanish Crescent" recipe.

IBM researchers have watched Briscolone work and how he goes about selecting ingredients to create a dish. These notes were fed through to Watson, who has studied over 20,000 recipes, data on food chemistry, and ratings from various people in categories like 'olfactory pleasantness.'

I.B.M.’s Watson beat “Jeopardy” champions two years ago. But can it whip up something tasty in the kitchen?

The pastry Watson had come up with was based on the criteria that it had to be unusual, healthy, and inspired by a Spanish cuisine. The ingredients of the pastry include cocoa, saffron, black pepper, almonds, honey, and replaces butter with vegetable oil instead.

Briscolone had to work these ingredients and adjust the portions to create the breakfast treat. He commented that, "If I could have used butter, it would have been a lot easier."

Michael Karasick, director of I.B.M.’s Almaden lab, tried the Spanish crescents for breakfast and said that they were "pretty good."
The potential business applications of QA are:

A. Customer care and help systems
B. Answering Healthcare questions
C. A & B
D. Playing Jeopardy, which can’t make IBM much money
E. None of the above
Computability and correctness of a computer program.
People try to design algorithms
e.g. to sort things, to run quickly
Algorithms

- Def. An algorithm is a precise, systematic process for an agent to produce a specified result.
- Programs are algorithms.
- Five properties characterize algorithms:
  - **Input specified** – tell form and amount of input required.
  - **Output specified** – tell form and amount of output produced.
  - **Definiteness** – say explicitly what to do & in what order.
  - **Effectiveness** – operations within agent’s abilities.
  - **Finiteness** – will stop and give an answer or say “none”.
Thinking About Computing

- Computers do things quickly ... except when they don’t
- Describing Computability
  - *Time proportional to X*
- Sometimes the time a computer takes is *linearly* proportional to how much data there is ...
  - Looking at every pixel in an image and changing it
  - Reading a CD
  - Making an image lighter in your photo software
  - Adding a column of numbers in a spreadsheet
  - Crawling the Internet looking for links
  - ... many more ... linear problems are common
Searching

- Guess a number between 1 and 100. How many guesses do you need?
- Scatter a deck of cards on the floor. How many do you have to turn over to find the ace of spades?
Linear = Time Proportional To N

Each item in a list that is N items long

- Problems whose work (computation time) is proportional to n are called n-time or linear time problems
  - Making an image lighter in your photo software
  - Adding a column of numbers in a spreadsheet
  - Crawling the Internet looking for links
  - ... many more ... linear problems are common

- Some problems are not ...
Time proportional to $n^2 = \text{Polynomial time}$

- Other computations have running time
  - proportional to $n^3$ – matrix multiplication
  - proportional to $n^4$
  - ...
- All of them are lumped together as “polynomial time computations”
  - Considered to be realistic ... a person can wait
  - Polynomial, but not linear
  - Many algorithms you would learn in 12B, 101
Finding an item in an unsorted list

What is the time complexity of finding an item in an unsorted list?

A. Less than linear (log(n) or constant)
B. Linear
C. Polynomial but more time than linear
D. NP (exponential like $2^n$)
Sorting

- Putting a sequence of items into alphabetical or numerical order
- First: let’s try exchanging them as we go along (bubbling them along)
  - *walrus seal whale gull clam*
- Algorithm: compare to all following items, reorder if needed
- Other ways to sort we will talk about in a minute
How Long To Sort w/ Bubble Sort?

- The pattern is, for n items
  - n-1 focus on first item in the list
  - n-2 focus on second item
  - n-3 focus on third item
  - ...
  - 1 on next to last
- n-1 rows in list, we just want
  a good estimate
- average of each row n/2,
- so (n-1) times (each row)
- Multiply by average = n/2
- = (n² – n)/2
- Computing time = Time proportional to n²

```c
Bubblesort (int data[],int n) {
    int tmp,i,j;
    for (i=0; i<n-1; i++)
        for (j=0; j<n-i-1; j++)
            if (data[j] > data[j+1]) {
                tmp = data[j];
                data[j] = data[j+1];
                data[j+1] = tmp;
            }
}
```
There are Different Algorithms

- Is there a better way to do sorting?

QUICKSORT

- Fastest known sorting algorithm in practice
- Average case: $O(N \log N)$ (we don’t prove it)
- Worst case: $O(N^2)$
  - But, the worst case seldom happens.
- A divide-and-conquer recursive algorithm

Video of selection vs quicksort: http://youtu.be/cVMKXXKoGu_Y
Quicksort is the best : Divide and Conquer

- Divide step:
  - Pick any element (*pivot*) \( v \) in \( S \)
  - Partition \( S - \{v\} \) into two disjoint groups
    \[ S_1 = \{x \in S - \{v\} \mid x \leq v\} \]
    \[ S_2 = \{x \in S - \{v\} \mid x \geq v\} \]

- Conquer step: recursively sort \( S_1 \) and \( S_2 \)

- Combine step: the sorted \( S_1 \) (by the time returned from recursion), followed by \( v \), followed by the sorted \( S_2 \) (i.e., nothing extra needs to be done)

To simplify, we may assume that we don’t have repetitive elements, So to ignore the ‘equality’ case!
Where does the logN come from?

Binary Search

- Pick a number between 1 and 1000. How many guesses will I need?

- Depends if it's sorted.
- Web Search: What about that index with 25 million entries?
- What is the “WORST CASE?”
Finding an item in a sorted list

What is the time complexity of finding an item in a sorted list?

A. Less than linear (log(n) or constant)
B. Linear
C. Polynomial but more time than linear
D. NP (exponential like $2^n$)
E. Unsolvable (like the halting problem)
Which grows slowest?

A. Time proportional to $N^2$
B. Time proportional to $N \times \log(N)$
To Infinity And Beyond

- There are more complex computations ...
  - Suppose you want to visit 28 cities in the US (for a rock concert?) and you want to minimize your how much you pay for airplane tickets
  - You could select an ordering of cities (SEA $\rightarrow$ PDX $\rightarrow$ SFO $\rightarrow$ LAX ...) and compute the ticket price.
  - Then pick another ordering (SEA $\rightarrow$ SFO $\rightarrow$ LAX $\rightarrow$ PDX ... ), compute this ticket price and compare to the previous one
  - Always keep the cheapest itinerary
- This seems very dumb ... is there a better way?
Traveling Salesman Problem

- Actually, no one knows a way to solve this problem significantly faster than checking all routes and picking the cheapest ...
- Not polynomial time ... so we are guessing that there is no polynomial solution (Non Polynomial = NP)
- This is what is called an NP-Complete problem
  - Many many related problems ... the best solution is “generate and check”
    - Best way to pack a container ship
    - Most efficient scheduling for high school students’ classes
    - Least fuel to deliver UPS packages in Washington
    - Fewest public alert broadcast stations for US
Which grows fastest?

A. Time proportional to $N^2$
B. Time proportional to $N \log(N)$
C. Time proportional to $N$
D. Time proportional to $2^N$
Summary: Computational Complexity

- Many computations have time proportional to $n$
- Many, like sort, have running time proportional to $n^2$
- Others have running time proportional to $n^3, n^4, ...$
- Some computations are computable in principle but not in practice: **NP-complete**
- Some things cannot be computed at all, such as the **Halting Problem**
Just a BIT more on colors!
White, Gray, Black

- You know that gray is just different degrees of white as the “light is turned down” till we get to black.

\[
\begin{align*}
\text{Black} &= [0, 0, 0] \\
\text{Gray} &= [128, 128, 128] \\
\text{White} &= [255, 255, 255]
\end{align*}
\]

White-gray-black all have same values for RGB.
Colors

Colors use different combinations of RGB

Purple
Red=160
Green=76
Blue=230
Random Numbers

- *Random numbers* should be called *random number sequences*, because the definition requires that no matter how many numbers you already know in the sequence, it’s not possible to predict the next one. A non-random sequence is 2, 4, 6, 8, 10, …

- Computers cannot produce random numbers (because computers are completely predictable), but they can produce a sequence of numbers that passes all of the tests for randomness. These are called *pseudo-random numbers*, but everyone drops the “pseudo” part.

- To generate a random number in Processing we write:
  
  ```
  random(<smallest possible number>, <largest possible number>).
  ```

- We get back a number – we can’t predict which – between the two limits, including the end points.

- To generate a random number between 0 and 255, write `random(0, 255)`.

- To generate a number between 0 and 1, write `random(0, 1)`.
What would we do in Processing? What does Random do?

```java
void draw() {
    mid_outer = lerpColor(outer, inner, .33);
    mid_inner = lerpColor(outer, inner, .66);
    fill(outer);
    rect(60, 60, 480, 480);
    fill(mid_outer);
    rect(100, 100, 400, 400);
    fill(mid_inner);
    rect(165, 200, 270, 260);
    fill(inner);
    rect(210, 260, 180, 160);
}

void mousePressed() {
    r = random(0,255);
    g = random(0,255);
    b = random(0,255);
    outer = color(r,g,b);
    r = random(0,255);
    g = random(0,255);
    b = random(0,255);
    inner = color(r,g,b);
}
```
void draw()
{
    mid_outer = lerpColor(outer, inner, .33);
    mid_inner = lerpColor(outer, inner, .66);
    fill(outer);
    rect(60, 60, 480, 480);
    fill(mid_outer);
    rect(100, 100, 400, 400);
    fill(mid_inner);
    rect(165, 200, 270, 260);
    fill(inner);
    rect(210, 260, 180, 160);
}

void mousePressed()
{
    r = random(0, 255);
    g = random(0, 255);
    b = random(0, 255);
    outer = color(r, g, b);
    r = random(0, 255);
    g = random(0, 255);
    b = random(0, 255);
    inner = color(r, g, b);
}
Positional Notation

- The RGB intensities are binary numbers
- Binary numbers, like decimal numbers, use *place notation*

\[
1101 = 1 \times 1000 + 1 \times 100 + 0 \times 10 + 1 \times 1
\]

\[
= 1 \times 10^3 + 1 \times 10^2 + 0 \times 10^1 + 1 \times 10^0
\]

except that the base is 2 not 10

\[
1101 = 1 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1
\]

\[
= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0
\]

1101 in binary is 13 in decimal
The Red of P As A Binary Number

Purple: Red=160, Green=76, Blue=230

Given a binary number, add up the powers of 2 corresponding to 1s

\[
\begin{align*}
1 \times 2^7 &= 1 \times 128 &= 128 \\
0 \times 2^6 &= 0 \times 64 &= 0 \\
1 \times 2^5 &= 1 \times 32 &= 32 \\
0 \times 2^4 &= 0 \times 16 &= 0 \\
0 \times 2^3 &= 0 \times 8 &= 0 \\
0 \times 2^2 &= 0 \times 4 &= 0 \\
0 \times 2^1 &= 0 \times 2 &= 0 \\
0 \times 2^0 &= 0 \times 1 &= 0 & \text{=160}
\end{align*}
\]
Green of P As A Binary Number

Purple: Red=160, Green=76, Blue=230

Given a binary number, add up the powers of 2 corresponding to 1s

<table>
<thead>
<tr>
<th>Binary</th>
<th>Decimal</th>
<th>Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0 \times 1</td>
</tr>
<tr>
<td>1</td>
<td>64</td>
<td>1 \times 64</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0 \times 32</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0 \times 16</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>1 \times 8</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>0 \times 4</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0 \times 2</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0 \times 1</td>
</tr>
</tbody>
</table>

0 1 0 0 1 1 0 0 = 76

What if we want it to be GREENER? BIGGER NUMBER OR SMALLER?
What if we want it to be GREENER? BIGGER NUMBER OR SMALLER?

A. BIGGER BINARY NUMBER for GREEN of RGB
B. SMALLER BINARY NUMBER for GREEN of RGB
What about making a redder Purple?

- So Purple is (160, 76, 230) which is
  \[
  \begin{align*}
  &1010 0000 \ 0100 1100 \ 1110 0110 \\
  &= 160 \ 76 \ 230
  \end{align*}
  \]

Suppose you decide it’s not “red” enough

- Increase the red by 16 = 1 0000
  \[
  \begin{align*}
  &1010 0000 \\
  + &\ 1 0000 \\
  &= 1011 0000
  \end{align*}
  \]

You already know how to add in binary, right?
How do we make a Redder Purple

- **ADD 16 more**

  \[
  \begin{array}{c}
  00110 000 \\
  1011 0000 \\
  + 1 0000 \\
  1100 0000 \\
  \end{array}
  \]

  Carries

  Original

  +16

  +16