There are 8 days left until turn-in for your pair-programming project
America’s First Six Programmers

Kathleen McNulty Mauchly Antonelli
Jean Jennings Bartik
Frances Snyder Holberton
Frances Bilas Spence
Marilyn Wescoff Meltzer
Ruth Lichterman Teitelbaum
Searching The WWW

Lawrence Snyder
University of Washington, Seattle
Looking In the Right Place

Google is not necessarily the first place to look!

- Go directly to a Web site -- www.irs.gov

Guessing a site’s URL is often very easy, making it a fast way to find information

- Go to your bookmarks -- dictionary.cambridge.org
- Go to the library -- www.lib.washington.edu
- Go to the place with the information you want -- www.npr.org

Ask, “What site provides this information?”
Google Advanced – Use It!

Advanced Search

Find pages with...

all these words:

this exact word or phrase:

any of these words:

none of these words:

numbers ranging from: to

Google
Boolean Queries

Search Engine words are independent

- Words don’t have to occur together
- Use Boolean queries and quotes
  - Logical Operators: AND, OR, NOT
    - grant AND wood AND american AND gothic
    - “van gogh” OR gauguin
    - vermeer AND girl AND NOT pearl
Searching strategies ...

- Limit by top level domains or format ... .edu
- Find terms most specific to topic ... ibuprofen
- Look elsewhere for candidate words, e.g. bio
- Use exact phrase only if universal, ... “Play it again”
- If too many hits, re-query ... let the computer work
- “Search within results” using “-” ... to get rid of junk
Google Advanced – Filtering

Find pages with...

all these words: 

this exact word or phrase: 

any of these words: 

none of these words: 

numbers ranging from: to 

Then narrow your results by...

language: 

region: 

last update: 

site or domain:
Queries, continued

Once found, ask if site is reliable source

- How authoritative is it? Can you believe it?
- How crucial is it that the information be true?
  - Cancer cure for Grandma
  - Hikes around Seattle
  - Party game
Is It REALLY True???

https://www.youtube.com/watch?v=CE0Q904gtMI
Is It REALLY True???
As you know, the Web uses http:// protocol. It’s 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: fonts, text color, headings, images, etc. etc. etc.
Plan: To Learn HTML Right Now

- In the next few slides I’ll blast through a quick overview of HTML & CSS, because …
  - We will need them for the next assignment
  - It’s much easier than Processing programming, so needs little explanation for “techies” like you
  - Makes the point that if you need to know something new about computing, you could teach it to yourself rather than taking a class …
  - There are loads of terrific tutorials online, so there’s a safety net
  - Copying someone else’s HTML is a noble tradition
Rule 0: Content is given directly; anything that is not content is given inside tags, like `<p>  </p>`

Rule 1: Tags made of `<` and `>` and used this way:

```
Attribute&Value

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

It produces: This is paragraph.

Rule 2: Tags must be paired or “self terminated”
Example

```html
<!doctype html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>All Downhill From Here</title>
  </head>
  <body style="background-color:black;
  font-family:helvetica;
  color:white">
    <h1>Downhill Skiing!!!</h1>
    <img src="skier.jpg" alt="Skier In Snow" />
  </body>
</html>
```

- Write HTML in text editor: notepad++ or TextWrangler
- The file extension is .html; show it in Firefox or your browser
Rule 3: An HTML file has this structure:

```html
<!doctype html>
<html>
  <head>
    <meta charset="utf-8"/>
    <title>Name of Page</title>
  </head>
  <body>
    Actual HTML page description goes here
  </body>
</html>
```

Rule 4: Tags must be properly nested

Rule 5: White space is mostly ignored

Rule 6: Attributes (style="color:red") preceded by space, name not quoted, value quoted
Three Sides: Basics of HTML

- To put in an image (.gif, .jpg, .png), use 1 tag
  `<img src="skier.jpg" alt="Skier in Snow"/>

- To put in a link, use 2 tags
  `<a href="http://www.cs.uw.edu/cse120">Pilot</a>

- Styling is specified with Cascading Style Sheets
- More on HTML & CSS (incl. good tutorials) at
  http://www.w3schools.com/html/default.asp
Larger Example

Paradoxes

Russell's Paradox

The Twentieth Century logician Bertrand Russell introduced a curious paradox: *This statement is false*. The statement can't be true, because it claims the converse. However, if it is not true, then it's false, just as it says. That makes it true. Paradoxically, it seems to be neither true nor false, or perhaps both true and false.

Magritte's Paradox

The famous Belgian artist René Magritte rendered the idea of Russell's Paradox visually in his famous painting *Ceci n'est pas une pipe*. The title translates from French, This Is Not A Pipe. The painting shows a pipe with the text *Ceci n'est pas une pipe* below it. Superficially, the painting looks like a true statement, since it is a picture of the pipe, not an actual pipe. However, the assertion is also part of the picture, which seems to make it false, because it is clearly a painting of a pipe. Paradoxically, the truth seems to depend on whether the statement is an assertion about the painting or a part of it. But, it's both.
<!doctype html>
<html>
<head>
  <meta charset="UTF-8"/>
  <title> Twentieth Century Paradoxes </title>
  <style>
    body {background-color: darkslategray;
         color: lightyellow}
    p {color: lightyellow}
    h1 {color: gold; text-align: center}
    h2 {color: darkorange}
    a {color: greenyellow}
  </style>
</head>
<body>
  <h1> Paradoxes </h1>
  <h2> Russell's Paradox </h2>
  <p> The Twentieth Century logician Bertrand <a href=""> Russell </a>
      introduced a curious paradox: <b style="color: red"> This statement is false. </b>
      The statement can't be true, because it claims the converse. However, if it is not true, then it's false, just as it says. That makes it true. Paradoxically, it seems to be neither true nor false, or perhaps both </p>
</body>
</html>
Larger Example

true and false. 

The famous Belgian artist René Magritte rendered the idea of Russell's Paradox visually in his famous painting *Ceci n'est pas une pipe*. The title translates from French, *This is not a pipe*. The painting shows a pipe with the text *Ceci n'est pas une pipe* below it. Superficially, this looks like a true statement, since it is a picture of a pipe, not an actual pipe. However, the assertion is also part of the picture, which seems to make it true, because it is clearly a painting of a pipe. Paradoxically, the truth seems to depend on whether the statement is an assertion about the painting or a part of the painting, or both.

Paradoxes

**Russell's Paradox**

The Twentieth Century logician Bertrand Russell introduced a curious paradox: *This statement is false*. The statement can't be true, because it claims the converse. However, if it is not true, then it's false, just as it says. That makes it true. Paradoxically, it seems to be neither true nor false, or perhaps both true and false.

**Magritte's Paradox**

The famous Belgian artist René Magritte rendered the idea of Russell's Paradox visually in his famous painting *Ceci n'est pas une pipe*. The title translates from French, *This is not a pipe*. The painting shows a pipe with the text *Ceci n'est pas une pipe* below it. Superficially, the painting looks like a true statement, since it is a picture of a pipe, not an actual pipe. However, the assertion is also part of the picture, which seems to make it false, because it is clearly a painting of a pipe. Paradoxically, the truth seems to depend on whether the statement is an assertion about the painting or a part of it. But, it's both.
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

- We’ll see how these work momentarily
How to crawl the Web:

- Begin with some Web sites, entered “manually”
- Select page not yet crawled; look at its HTML
  - For each keyword, associate it with this page’s URL
  - Harvest words from URL and inside <title> tags ...
  - For every link tag on the page, associate the URL with the words inside of the anchor text, that is,

- Save all links and add to list to be crawled

http://www.google.com/insidesearch/howsearchworks/thestory/
Crawling Pages Builds Index Data

Index

- a:
  - ...  
- carton:  
  - pet-h
- cat:  
  - www
- eye:  
  - www
- green  
  - www
- head  
  - pet-h
- milk  
  - pet-h
- zzzzzzz

URL: www.fan.cy/beckyR

Green Eye Cat

My Cat Molly

URL: pet-home.com/molly

LOL Alert: See Molly with her head in a milk carton
Net Result From Crawling A Page

- Build an index
- Terms on a page are not all equally useful:
  - Anchors from other pages
  - Terms in URL, esp. path items
  - Title
  - H1
  - H2
  - Meta description
  - Alt helps with images
When crawling’s “done” (it’s never done), the result is an *index*, a special data structure a query processor uses to look up queries:
Google has never revealed all details of the ranking algorithm, but we know ...

- URL’s are ranked higher for words that occur in the URL and in anchors
- URL’s get ranked higher if more pages point to them, it’s like: A links to B is a vote by A for B
- URL’s get ranked higher if the pages that point to them are ranked higher
Virtual Folders are a “crawling/querying” technology that helps you
- Mac: Smart Folders
- PC: Saved Folders

In both cases your files are “indexed”, that is, crawled, and the query you make results in a hit list that can become a “smart” folder

It’s like Googling the stuff on your own computer
Query “thesis”

- Hit List for this machine
Search Engines … A Summary

- A search engine has two parts
  - Crawler, to index the data
  - Query Processor, to answer queries based on index

- In the case of many hits, a query processor must rank the results; page rank does that by
  - “using data differentially” ... not all associations are equivalent; anchors and file names count more
  - “noting relationship of pages” ... a page is more important if important pages link to it

Google, Bing, Yahoo and other Search Engines Use All of These Ideas