

### **Announcements (0:00 – 6:00)**

- We are holding over a hundred hours a week of office hours, which should be plenty. Please do not expect a teaching fellow to spend more than 2-5 minutes with you at office hours.

### **Huffman Coding (6:00 – 12:00)**

- You can use binary tree to implement an algorithm designed by Huffman to compress a file.
- It attempts to use fewer than 8 bytes per character to store the file by taking advantage of redundancies
- Here is the algorithm:
  - Make a table showing frequency of each character in the file
  - Make a node corresponding to each distinct character. Assign to each its frequency in the file. Place all of these nodes in a “forest.”
  - Sort these nodes by frequency.
  - Combine the nodes with the lowest frequencies by making them the children of a parent whose number is the sum of the frequencies.
  - Put the parent back in the forest.
  - Go back to step 4 and repeat, until there is one node remaining in the forest
  - The binary representation of any character is then computed as follows: determine the series of lefts/rights to get to the character from the root, then replace “left” with 0 and “right” with 1
- The result of this algorithm is that very frequently appearing characters have shorter binary representations than rarely appearing characters.

### **The Internets (12:00 – 34:00)**

- The Internet is a network of networks
- You plug into a home router, and this router, say, connects to a Cable modem, or a data jack in the wall, or a wireless access point in the wall
- Every computer is assigned a unique IP address of the form 1.2.3.4
- A router route data from one direction to another. Given data destined for a particular IP address, a router can determine which direction to send it.
- You can use UNIX command `nslookup <domainname>` to determine the IP address of the server for that domain name
- If we do this for `cnn.com`, and then type that IP address into Firefox, it still takes us to `cnn.com`
- (Actually, websites can have multiple IP addresses, and `cnn.com` does. This is because they have multiple servers.)

- You can use the command `traceroute <domain name>` to determine the series of computers (indicated by their IP addresses) required to get from your computer (or the cloud if that's where you're working) to the server of that domain name
- For us, the Internet will serve as an *infrastructure* on which we will write code
- TCP/IP is the protocol according to which computers are expected to communicate, standardized so that they can understand one another
- This works at a very low level, dealing with to/from addresses, etc.
- We can think of HTTP as sitting on top of TCP/IP. It is a protocol which specifies higher level details, such as what kind of data will be exchanged, what the data should look like
- On top of HTTP are application layer languages, such as HTML and PHP
- You can see an example of HTML by going to any webpage and choosing View>Page Source in your browser

### Creating a Webpage (34:00 – 44:00)

- On slide 12 we see an example of an extremely simple webpage
- A tag is what you use to mark up your text. Tags begin with an angle bracket, have a keyword, and close with an angle bracket. For instance, `<head>`
- The rule in XHTML is that every open tag `<keyword>` must be matched with a close tag `</keyword>`
- We use tags to mark the head and the body
- The header will contain things like the title
- Using head and body tags we prepare a simple webpage with only a title and some text and call it `index.html`
- When we first try to view this webpage, we get error 403. This means the page is forbidden.
- We can use the `-l` flag on `ls` to see permissions
- Upon listing the contents of `public_html` with permissions, we see that the `index.html` has permissions `"rw-----"` What does this mean?
- Permissions are indicated by three octal numbers which represent, from left to right, the owner's, group's, and world's permissions
- Within each octal number, the three bits represent permission to, from left to right, read, write, and execute the file.
- So 100 means permission to read, 101 means permission to read and execute, 110 means permission to read and write, etc.
- Note that Linux displays these bits using letters and dashes, to make it easier for us. So 111 is displayed as `rxw`, 000 is represented as `---`.
- Right now the permissions on `index.html` are 110 000 000 (displayed by linux as `rw-----`). This means the owner has permission to read and write only, but world has no permissions.
- We want world and group to be able to read the file so that people can view `index.html` in their browser. This means we want permissions to be 110 100 100 (or `rw-r--r--`)

- We achieve this by saying “chmod 644 index.html” because in octal, 110 is 6 and 100 is 4
- We also must make the directory world executable so we additionally call “chmod 755 .”
- This allows the world to open the directory in their browser

### **XHTML (44:00 – 69:30)**

- Tags can also have attributes. For instance, we can set background color using an attribute in the body tag: `<body bgcolor = “yellow”>`
- Suppose we want to center the text in the body. We can accomplish this by putting the text in `<center>` tags
- In HTML, whitespace is ignored. (That’s why we can’t center the text by holding space bar a while)
- So if we want to put two lines of text separated by a line of whitespace, we use `<br/>` because the whitespace produced by just pressing enter would be ignored
- `<br/>` is shorthand for `<br></br>`
- Make the heading big and bold by putting it in `<h1>` tags
- If we want to be more sophisticated, we can put the heading in `<div>` tags and include a style attribute to specify font size and color
- We include an image using the `<img>` tag and src attribute: `<img src = “myimage.jpg”>`
- This has to be chmod-ed too!
- Take note of the three lines at the top of the file shown in slide 12. You should include these lines at the top of any XHTML file you write.
- In order for XHTML source to be considered “valid”, it has to adhere to certain standards. You can check to see if your code is valid using <http://validator.w3.org>
- Note that HTML is a markup language, not a programming language.
- This means it tells the browser what to do structurally or aesthetically, but contains no logic
- One useful thing we can do structurally is make a table, to help lay things out properly
- Cascading style sheets (CSS) are useful to fine tune appearance of your webpage, and to achieve consistency

### **Fake Google (69:30 – 76:30)**

- Let’s look at Google as an example of a really simple webpage
- A form is something with fields, buttons, checkboxes, and or menus
- The main part of the Google homepage is the form that allows you to enter a search term and press search
- A form is indicated using a `<form>` tag
- The form tag has an action attribute which indicates where you want to send the contents of the form when it is submitted

Week 8 Monday  
November 10, 2008  
Scribe Notes

Computer Science 50  
Fall 2008  
Anjuli Kannan

- Right now action is set to “/search”, meaning [www.google.com/search](http://www.google.com/search)
- We can copy and paste the form into our own homepage
- We need to change the action to be the full path
- The result is our own “fake Google”