Recursion and Recursive Structures
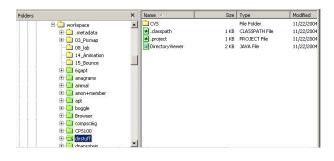
- **Definition in the dictionary: in mathematics an expression in which a value is calculated by using preceding terms of the expression**
  - **Pretty worthless, doesn't convey details or power**

- **Consider folders and files in a computer, what's in a folder?**
  - **Files, other folders, anything else?**
  - **Is there a distinction between a folder and a file?**

- **Viewpoint is everything, who is the viewer? The viewee?**
  - **What does this have to do with physics?**
  - **How do we look at files via programming**

Files and Folders: Files and Directories

- **OS X view of *Directory Structure***

Different OS, Same View

- **Windows 2000 Files and Directories**

---

java.io.File

- **An abstract representation of file and directory pathnames**
  - **C:\eclipse\workspace\dirstuff\DirectoryViewer.java**
  - **/Users/ola/Desktop/eclipse/workspace/dirstuff/DV.java**

- **What is a pathname?**
  - **Given the path, can you find/navigate to the file/folder?**
  - **Is there more than one way to get to a folder? Why?**

- **What accessor methods exist for Files/Directories**
  - **getName()**
  - **length(), lastModified()**
  - **isDirectory(), isFile()**

Problem? How 'large' is a directory?

- The size accessor method and file-system view tells how much diskspace the directory entry needs
  - This is small, it requires name, date, list-of-files
  - The list-of-files isn't really part of the directory size, but it doesn't matter, we want SIZE

- How does the 'search' command work to find a file?
  - Search everything from scratch each time
  - Cache results with indexing to avoid re-searching

- How do interact programmatically with file system in platform independent way?
  - Need an abstraction (hint: see java.io.File)

Code to visit a Folder (dirstuff)

```java
public void visit(File f, int level){
   if (f.isDirectory()){
      System.out.println(tab(level)+f.length()
                         + " **** " + f.getName());
      File[] files = f.listFiles();
      for(int k=0; k < files.length; k++){
         visit(files[k],level+1);
      }
   }
   else {
      System.out.println(tab(level)+f.length()
                         + "\t" + f.getName());
   }
}
```

Reasoning about File/Directory code

- What is the purpose of the method `visit`?
  - What does `visit` do?
  - What methods does `visit` call?
  - How does `visit` use `java.io.File`?

- Does the method call itself?
  - No, but it calls a method named visit, with a different parameter
  - Think of hundreds of little visit clones, do some work, pass other work off to clone
  - How can we modify the method to print directory last?
  - How can we modify method to calculate "real" size?

Who is Alan Perlis?

- It is easier to write an incorrect program than to understand a correct one
- Simplicity does not precede complexity, but follows it
- If you have a procedure with ten parameters you probably missed some
- If a listener nods his head when you're explaining your program, wake him up
- Programming is an unnatural act
- Won first Turing award

`http://www.cs.yale.edu/homes/perlis-alan/quotes.html`

What is recursion (revisited)?

- Structure in which a container can hold/use other containers
  - A directory holds file and directories
  - A Russian doll holds other russian dolls?
  - A method that calls 'itself' or clones

- Do russian dolls nest infinitely?
- Do folders hold infinite subfolders?
- Can a method call methods infinitely?

- We need a way out of the mess
  - Last doll, directory with no subdirectories
  - Method that makes no calls

Mathematical Definition of Factorial

- 6! = 1 x 2 x 3 x 4 x 5 x 6
- What's n!
  - If n equals 1 or 0, then it's 1
  - Otherwise it's n x (n-1)!

- Does this get the job done? Is it constructive?
  - What does the method below return for 5?
  - What about –1?

```
public static int factorial(int n){
   if (n == 1 || n == 0) return 1;
   else return n * factorial(n-1);
}
```

Recursive Structure and Methods

- There must be some way out of the recursion
  - A structure with no substructure
  - A method with no recursive method calls
  - This is the *base-case* of the recursion

- The substructure must be similar to the original structure
  - Russian dolls don't hold machine guns (recursively)
  - Each russian doll holds a smaller doll
  - Each call of `visit()` gets closer "to the bottom"
  - Each call of `factorial()` gets closer to 0 or 1
  - Ensure the base case is eventually reached

Recursion and Bioinformatics

- Methods we "studied" for sequence alignment
  - How do we align sequences optimally?
  - What about lots of sequences?
  - Dynamic programming is related to recursion (often implement DP recursively)

- How do we find CG rich regions in a sequence?
  - Divide sequence in half, look for CG rich regions in each half and the overlap of halves
  - How do look for CG rich regions in halves?

Fred Brooks

- **… on computing pioneer Howard Aiken** *"the problem was not to keep people from stealing your ideas, but to make them steal them."*
- **Duke valedictorian 1953, started UNC Computer Science Dept in 1964, won Turing Award in 1999**
- **Mythical-Man Month,** *"Adding man-power to a late project makes it later", … "There is no silver-bullet for Software Engineering… [because of essential complexity]"*
- **Highest paid faculty in UNC Arts and Sciences**