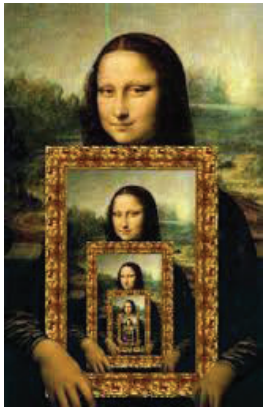
Compsci 101
How Dictionaries work, Recursion



Susan Rodger
April 14, 2022

`W` is for …

- **World Wide Web**
  - Where http meets tcp/ip?
- **WiFi**
  - We need and use this every day
- **Windows**
  - From OS to ...

`X` is for …

- **XOR**
  - (a or b) and not (a and b), a.k.a. symmetric difference
- **XML**
  - eXtensible Markup Language
- **Xerox Parc**
  - From Mice to Windows

Rediet Abebe

- CS PhD ‘19 Cornell
- Harvard Fellow til ‘22
- UC Berkeley Assist. Prof
- Research: AI, Inequality and Social Impact
- Co-founded Black in AI
- Co-founded Mechanism Design for Social Good

“For the most part, algorithms didn’t create inequity and inequality, but the fact that we didn’t have people who were engaging with algorithms’ role was exacerbating this existing inequality. With any sort of social issue, an algorithm can make things a lot worse, or it can help you understand what’s going on better and try to move things in a positive direction.”

Announcements

- APT-8 due today!
- Assign 6 Recommender, due Tuesday 4/19
  - One grace day, NO LATE DAYS! NO Extensions!
  - MUST TURN in BY Wednesday 4/20!!
- Assign 7 Create due, Wednesday, April 20!
  - Grace period with no penalty thru Sunday, April 24
  - No Late turnins or extensions after April 24!
- Lab 12 Friday, do prelab!
- Final Exam - 9am, Wednesday, April 27
  - 3 hours, in person, covers topics through last day

Assignment 7: Create, Due Apr 20

Grace period til Apr 24, No late days after Apr 24!
This assignment is **required**!

Pick one:

Video: Green dance, advertisement for 101, song, other
Poem or Multiple Haikus
Story
Comic
One-pager
Feedback

Let's see some examples

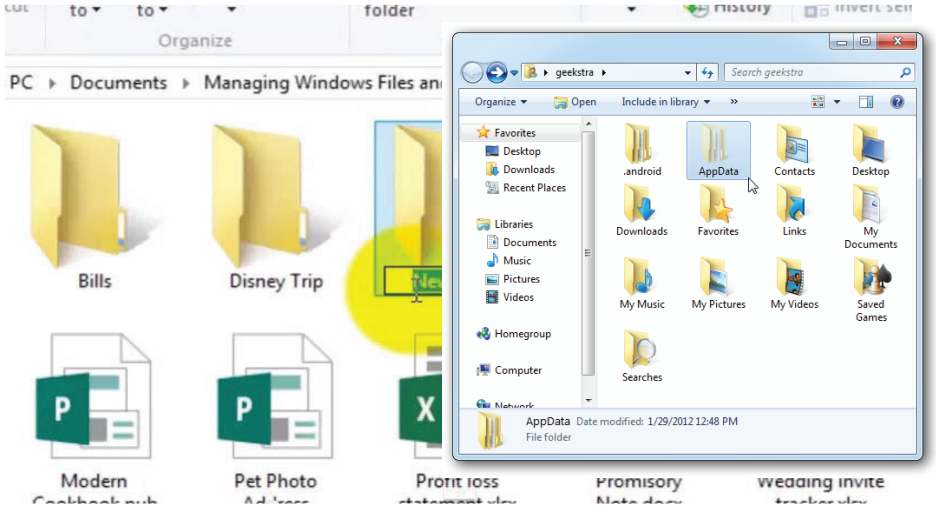
PFTD

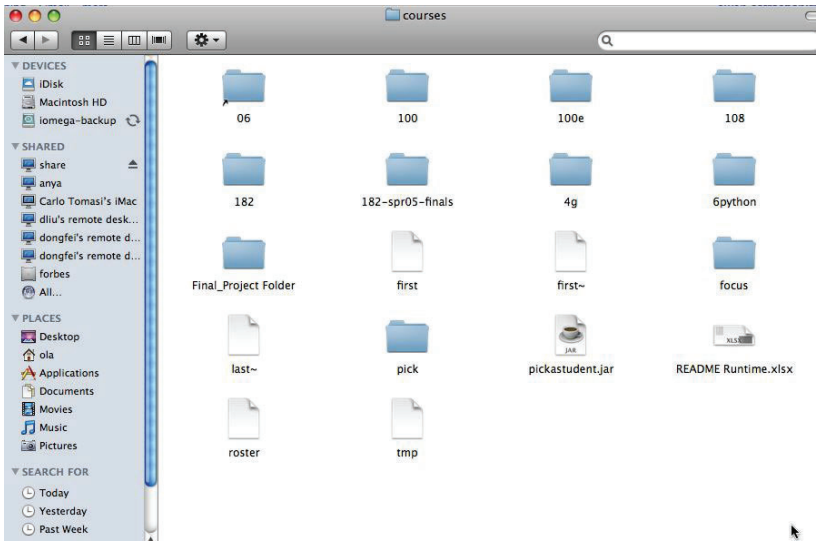
- Recursion
  - Technique for solving a problem by solving smaller problems

Recursion

- Solving a problem by solving similar but smaller problems

What's in a file-system Folder?

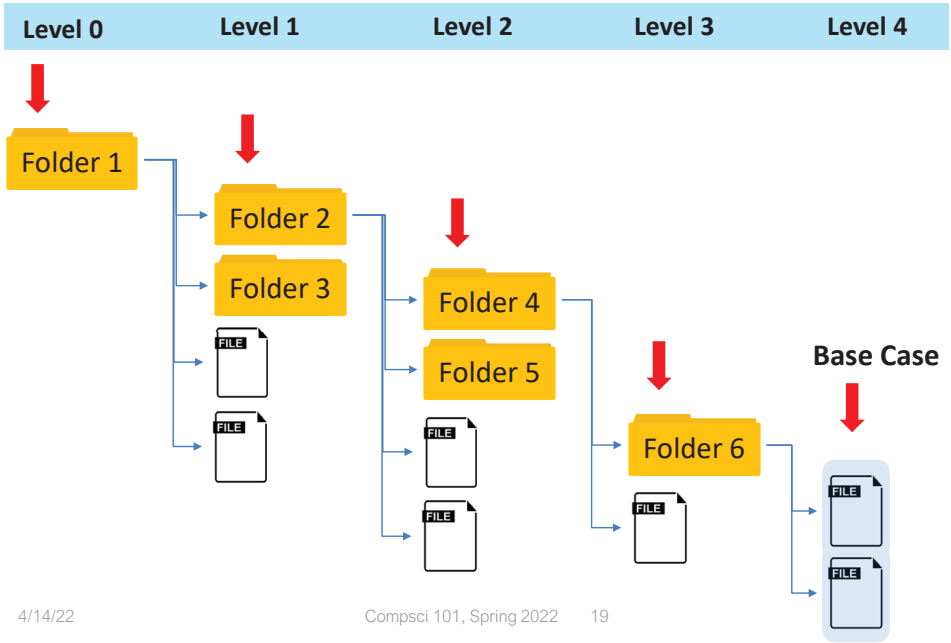




What's in a folder on your computer?

- **Where are the large files?**
- **How do you find them?**
- **They take up space!**
  - What's the plan –
    1. Erase?
    2. Backup?

Hierarchy in Folder Structure



Recursion (idea) to print ALL files in a folder

- **A folder can have sub folders and files**
- **A file cannot have sub files**

```
def visit(dirname):
    for inner in dirname:
        if isdir(inner):          <-- Is that a directory?
            visit(inner)
        else:                     <-- If not a directory, it will be a file
            print(name(inner), size(inner))
```


Finding large files: FileVisit.py

```
def bigfiles(dirname,min_size):
    large = []
    for sub in os.listdir(dirname):
        path = os.path.join(dirname,sub)
        if os.path.isdir(path):
            subs = bigfiles(path,min_size)
            large.extend(subs)
        else:
            size = os.path.getsize(path)
            if size > min_size:
                large.append((path,size))
    return large

# on Mac like this:
#bigs = bigfiles("/Users/Susan/Documents",10000)
# on Windows like this:
bigs = bigfiles("C:\\Users\\Susan\\Documents",10000)
```


Example Run

- ('C:\\Users\\Susan\\files\\courses\\cps101\\workspace\\spring2015\\assign4_transform\\data\\romeo.txt', 153088L)
- ('C:\\Users\\Susan\\files\\courses\\cps101\\workspace\\spring2015\\assign4_transform\\data\\twain.txt', 13421L)
- ('C:\\Users\\Susan\\files\\courses\\cps101\\workspace\\spring2015\\assign5_hangman\\src\\lowerwords.txt', 408679L)
- …

Finding Large Files questions
bit.ly/101s22-0414-1

The os and os.path libraries

- Libraries use an API to isolate system dependencies
  - C:\\x\\y # windows
  - /Users/Susan/Desktop # mac
- FAT-32, ReFS, WinFS, HFS, HSF+, fs
  - Underneath, these systems are different
  - Python API insulates and protects programmer
- Why do we have `os.path.join(x,y)`?
  - x = /Users/Susan/Documents
  - y = file1.txt
  - Output = /Users/Susan/Documents/file1.txt

Dissecting FileVisit.py

- **How do we find the contents of a folder?**
  - Another name for folder: directory
- **How do we identify folder? (by name)**
  - os.listdir(dirname) returns a list of files and folder
- **Path is c:\user\rodger\foo or /Users/rodger/bar**
  - os.path.join(dir,sub) returns full path
  - Platform independent paths
- **What's the difference between file and folder?**
  - `os.path.isdir()` and `os.path.getsize()`

Does the function call itself? No!

```
def visit(dirname):
    for inner in dirname:
        if isdir(inner):
            visit(inner)
        else:
            print(name(inner), size(inner))
```

- **Is a file inside itself? No!**
- **Does pseudo code make sense?**
  - Details make this a little harder in Python, but close!

Structure matches Code

Find large files

If you see a folder,

1. Find the large files and subfolders
2. For the subfolders, repeat the process of finding large files and any other folders within that subfolder
3. Repeat the process until you reach the last folder

Compress or Zip a folder

If you see a folder,

1. Find the files and subfolders
2. For the subfolders, repeat the process of finding files and any other folders within that subfolder
3. At the last stage, start compressing files and move up the folder hierarchy

Structure matches Code

- Structure of list of lists
  - Can also lead to processing a list which requires processing a list which …
- Is e in this list?
  - How many lists do you have to look in?

```
[ [ [a,b], [c,d] ], [a, [b,c],d] ]
```


Structure matches Code

- Structure of expressions
  - Can also lead to processing an expressions which requires processing an expression…
- How do you evaluate expression?

```
(a *(b + c (d + e*f)) + (a* (b+d)))
```


Recursion Summary

- Make Simpler or smaller calls
  - Call a clone of itself with different input
- Must have a base case when no recursive call can be made
  - Example - The last folder in the folder hierarchy will not have any subfolders. It can only have files. That forms the base case

Mystery Recursion
bit.ly/101s22-0414-2

Something Recursion
bit.ly/101s22-0414-3

Recursion in Pictures

- http://xkcd.com/543/