# PFtHWBTh

- **Discuss midterm, class expectations, dictionaries**
  - What's the purpose of a midterm, how to achieve?
- **Clever Hangman: Design and Implementation**
  - How to debug, antibugging, examining state
  - How do we know the program is working?
- **Self-reference, regular expressions, recursion**
  - Tools that can help solve problems
  - Beauty is in the eye of the beholder

## Being Clever at Games, aka Cheating

- When you play solitaire, does the computer cheat?
  - What would this mean?
- When you play chess, who wins?
  - Computer? You?
- When you play Poker?
  - Online Gambling
  - Poker/US Laws
- Hangman???



## IBM Watson

http://ibm.co/godoMD

Watson was optimized to tackle a specific challenge: competing against the world's best Jeopardy! contestants


Beyond Jeopardy!, the IBM team is working to deploy this technology across industries such as healthcare, finance and customer service.



## Computer is clever at Hangman?

- Cheating/clever NOT OK! : user guesses 't'
  - Computer says 'no occurrence of t'
  - Later secret word is 'chocolate'
- Clever IS OK: user guesses 't' knowing that word is six letters with third letter 'r': _ _ r _ _ _
  - Computer's secret word is 'street'
  - Computer says 'no occurrence of t'
    - Changes secret word to 'person'
  - Why is this ok?



## How to program clever game?

- Suppose the possible words are those here:

```
[ "OBOE", "NOON", "ODOR", "ROOM", "TRIP",
  "SOLO", "PICK", "FRAT", "HOOP"]
```

- What happens if player guesses 'O' as the very first guess?
  - What should computer's secret word be?
- ["OBOE", "ODOR"] is an *equivalence class*
- ["NOON", "ROOM", "HOOP"] is too
- What about words with no O's?

## Aside: Help in debugging programs

- Often very useful to print information
  - Contents of list, set, dictionary, …
  - After computer winnows possible words?
  - After computer creates dictionary?
- We don't want debugging/print statements in released (final) code

```
if _DEBUG:
   for key in d:
       print key,d[key]
```



## How do we use global _DEBUG?

- If it's a global variable we don't need to declare it when we use it in a function when …
  - If we write/change global variable must declare
  - If we read/access, don't need to declare! Use this!
- Sometimes useful to create print/debug function

```
def debug(info):
    if _DEBUG:
        print info

debug("%s has len %d" % (key, d[key]) )
```



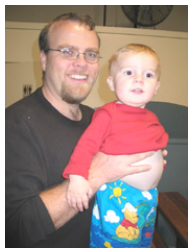
## Clever and Python reminders

- Dictionary keys must be immutable
  - Strings are good, lists are bad, tuples are ok
  - When can you write s[i] = 'c'?
- Write a small amount of code and test it
  - Knowing where your bug is beats trying to find it
  - This is a skill you'll get much better at, but practice*3
- This program combines … lists, dictionaries, sorting (or max), strings,
  - Plus it's both frustrating and fun



## PHP, Rasmus Lerdorf and Others

- Rasmus Lerdorf
  - Qeqertarsuaq, Greenland
  - 1995 started PHP, now part of it
  - http://en.wikipedia.org/wiki/PHP
- Personal Home Page
  - No longer an acronym



- "When the world becomes standard, I will start caring about standards."

  Rasmus Lerdorf

## Grammars, Regex, Problems and More

- Grammars are used
  - In computer science, designing software and hardware
  - In English, in Spanish, in all *natural* languages
  - In genomics, grammar of DNA?
- *Regular Expressions* math, compsci, real problems
  - How do recognize SPAM? Part statistics, part regex
  - How do we tell if email address entered is valid?
  - How do we search with wild-cards, e.g., *@duke.edu
- How do we recognize a valid Python program?



## Grammars and Regex

```
<integer>  ::= <digit> | <digit> <integer>
<digit> ::= 0| 1| 2| 3| 4| 5| 6| 7| 8| 9
```

- Why is 1234 a valid integer? Is 01234 a valid integer?
  - How could we avoid leading zeros?
  - What about a floating point number?
- Regular expressions: mathematical and applied
  - Create regexps from . + * ( | \ $
  - Understanding how these work best done by example
    - [A-Za-z]+\.[A-Za-z]+@ and then more
  - http://pdos.csail.mit.edu/scigen/



## Regular Expressions

- a, a+, a*, [abc], [a-z], ^a, a$
  - These are useful in and of themselves, madlibs, RSG
  - Also good for exploring problems and grammars
- Parsing and handling HTML
  - Finding <a href="http://...
  - Why is this useful to Bing | Google?
- Lots of details, more of a preview
  - Where to look for more info?




## Compsci 101: Re[gex|cursion]

- Recursion: self-referential structures and code
  - Look up recursion in [Google | Bing]
  - Look-it up in the index of ...
  - What is a folder on your computer's desktop?
- Powerful tool for elegantly expressing algorithms
  - Never necessary, but alternative can be hard to develop, lengthy, tricky, … (but then again …)
  - Part of essential toolkit of computer scientist
    - Arguably not essential for web developer, entrepreneur, social media promoter, …

## What's the deal with self-reference?

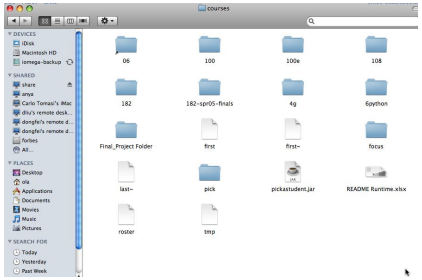
```
def visit(dirname):
    for file in dirname:
        if isdir(file): visit(file)
        else: print file
```

- Does pseudo code make sense?
  - Details make this a little harder in Python, but close!
- Recursive functions
  - Function doesn't call itself, each function is a separate "thing", with its own state
  - Must have a *base case*, no recursive calls made, no self-referential work done



## What's in a folder on your computer?

- Where are the large files? How do you find them?
  - Can a folder be inside a folder? Why?





## Finding large files: FileVisit.py

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

bigs = bigfiles("c:\Users",10000)
[(file,102030),(nfile,1030303),(pfile,10001)]
```



## 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 …
  - Path is c:\user\ola\foo or /Users/ola/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()`

## Creativity with self-reference

- Sometimes madlibs are fun (corollary?)
  - Humans fill in the blanks
  - Computers automatically fill in the blanks

The <apt-name> APT was really <description> but I didn't do it because I <excuse>

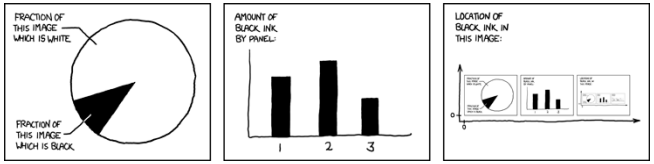
<description> :: "cool", "terrible", "baller", ...
<excuse> :: "was too tired", "didn't know how",...
<excuse> :: <excuse> and <excuse>

- See SimpleGrammar.py

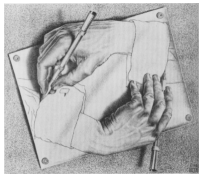


## Recursion in Pictures

- http://xkcd.com/688/ and http://xkcd.com/543/









## The power of regular expressions

- Interdisciplinary:
  - Music and Compsci

(for Compsci 108 final project)

- Who is Ge Wang?

http://www.youtube.com/watch?v=ADEHmkL3HBg

The final product is so much more than we had hoped for though it was something that we aimed for from the beginning.

Our investment into a huge and meticulous design process was a huge factor in making later progress. 35000+ lines of code / design / documentation gave us a project we were all very happy and proud to be a part of.



## Grammars for fun and recursion

- http://en.wikipedia.org/wiki/SCIgen
- http://www.elsewhere.org/pomo/
- http://www-cs-faculty.stanford.edu/~zelenski/rsg/
- I need an extension because <plea>.
- <plea>::
  - <dubious-excuse>,
  - <dubious-excuse> and <plea>