Plan for October 29

- **Review dictionaries and their use**
  - Very efficient, easy to use
  - Efficiency doesn't matter much for small data
  - Programmer time, how expensive is it?

- **Review APTs, reminder about APT quiz**
  - Quiz must be done alone, we don't look at code
    - But we could look at code to ensure no copying!
  - Quiz will be mostly straightforward application
    - If you're up-to-speed on APTs this week, good!

A Python view of dictionaries

- A collection of (key,value) pairs that is similar syntactically to a list
  - A list can be accessed by index: a[3]
  - A dictionary can be accessed by key: d["cat"]

- The key in a dictionary must be immutable
  - Essentially because key converted to number and number used as index (to find value)

- Finding the value associated with a key is very fast
  - Essentially doesn't depend on # keys!

Python syntax for dictionaries

- **Coding with Dictionaries**
  - **Error to access d[key] for update if key not in d**

Dictionary Syntax/Function	Meaning
`d.items()`	List of (key,value) tuples
`d.keys()`	List of keys
`d.values()`	List of values
`d.get(key)`	Like d[key], no error
`d`	Query like d.keys()

Case Study: Counting # occurrences

- See Counter.py, what does function `countup` return? Conceptually?
  - Words is a list of strings
  - Sorting tuples looks at first element, breaks ties with second `("dog",2) > ("cat",4)`

```
['dog','cat','bug','cat','dog','cat','cat','bug']
```

```
def countup(words):
    pairs = [(w,words.count(w)) for w in words]
    return sorted(set(pairs))
```

Counting more quickly

- **What makes `countup` "slow"?**
  - Why is a set returned? Why a sorted set?
  - How many times is `words.count(w)` called?
  - Making countup faster vs. a new approach
    - Let's use a dictionary!

```
def countup(words):
    pairs = [(w,words.count(w)) for w in words]
    return sorted(set(pairs))
```

Counting more quickly

- Easy to code, use words.count! But after counting 'dog', we count 'cat', and then …
  - Look at a million words in counting a thousand
  - Instead, look at words once! Update per-word counter, so much faster with dictionaries!

```
def fastcount(words):
    d = {}
    for w in words:
        if w in d:
            d[w] += 1
        else:
            d[w] = 1
    return sorted(d.items())
```

Answer Questions

http://bit.ly/101fall15-oct29-1

danah boyd

Dr. danah boyd is a Senior Researcher at Microsoft Research, … a Visiting Researcher at Harvard Law School, …Her work examines everyday practices involving social media, with specific attention to youth engagement, privacy, and risky behaviors. She heads Data & Society (Research Institute) and recently authored *It's Complicated: The Social Lives of Networked Teens.*

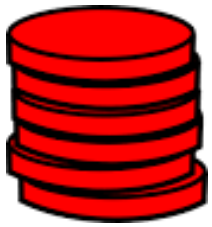
"we need those who are thinking about social justice to understand technology and those who understand technology to commit to social justice."

http://bit.ly/1GuB9x2

Solving APTs

http://www.cs.duke.edu/csed/pythonapt/networth.html

- If Harry pays Sally $10.23,
  - "Harry:Sally:10.23" and Harry is out $10.23
- Given a string in this form, how do we extract payer, payee, amount?
  - Conceptually
  - In Python



After extracting transaction info …

- Why is a dictionary useful? What are (key,value) pairs?
  - Think about how to do this by hand, keep a sheet with each person's name, update the amount next to their name
  - Look up name, get amount
- General dictionary update methods
  - Check if key seen before, update `d[key] +=`
  - If not seen, initialize first time, `d[key] =`

So many APTs have this format!

- Initialize structure before looping over data
  - List, set, string, dictionary
- Loop over data and update structure
  - Extract info from element, update by .add, .append, +=, etc.
- May need to process structure for return
  - Sort, remove some, change format, etc.
  - What does `d.items()` return for a dictionary?
    - List of (key,value) tuples!

Finishing up VenmoTracker

- Value stored in dictionary before return

```
[('drew', 10.0), ('owen', -30.0),
 ('robert', 10.0), ('susan', 10.0)]
```

- How do access name and amount in each tuple? How do we loop over tuples?
- How do we create a string from a string and a float?
- How do we sort, when do we sort?

Answer Questions

http://bit.ly/101fall15-oct29-2

Member Club APT

http://www.cs.duke.edu/csed/pythonapt/membercheck.html

- Given two lists A and B, how can you find a list of values in both lists?
  - `for x in A: if x in B:`
  - `both = list(set(A)& set(B))`

- Ideally you'll see the set solution quickly, but solving the problem is important!

DictionaryTimings.py

- Updating (key,value) pairs in structures
  - Search through unordered list
  - Search through ordered list
  - Use dictionary

- Why is searching through ordered list fast?
  - Guess a number from 1 to 1000, first guess?
  - What is 2^{10}? Why is this relevant? 2^{20}?
  - Dictionary is faster! But not ordered