Plan for TDAFB (after fall break)

- Hear via email about plans to catch-up and stay-ahead
  - Profs. Rodger and Astrachan to supply info

- Review APTs due next week
  - Ideas, Code, Algorithms, approaches
  - Review loops and structures
  - How sets can be useful

- Word games to start things off
  - Assignment coming
Succeeding with APTs in Compsci 101

● First do example, solve problem by hand, then Python
  - Capable of doing all examples by hand for APT
    - You don't want to have to do *all* of them by hand!

● Using built-in functions:
  - len(x), sorted(x), max(x), ''.join([...])
  - What is return type of each, which is "it depends"

● Two loops (index/element), Three structures
  - String, list, and now set
Computational Jumble

Use this problem to think about word games

- Human approach
  - What do you do?

- Computational method?
  - Cheating or insight?

http://www.jumble.com

Yesterday's Jumbies: SKUNK TWEAK GAMBOL ADJOIN
Answer: The plastic surgeon sought the clerk's help because she had a "KNOWS" job
Eros by any other name would be sore

- See Jumble.py (accessible via snarf)
  - Find standard or canonical form for jumbled
  - "aeprs" is ... in alphabetical order!
  - `sorted('spear')` and `sorted([3,1,2,8])`

- Review of list comprehensions,
  - Given a large list of strings named words, what does this comprehension do?

  ```python
  [w for w in words if len(w) == len('rose')]
  ```
How long does this take in Jumble.py?

[w for w in words if sorted(w) == sorted(jword)]

- Go through each of 354,984 words and ...
  - Sort the word, which returns a list
  - Compare to sorted form of each word

- What if we change boolean filter?
  - Why does this appear to make a difference?

if len(w) == len(jw) and sorted(w) == sorted(jw)
Jumble programming concepts

● When you run the program it starts in __main__
   - This is how Python works, boilerplate code

● What's the variable words at beginning?
   - Global variable. Accessible in every function in the Python module
   - Used sparingly can be useful in a small module
   - Abused, can lead to hard to maintain code
Answer Questions


After questions, solving APTs: ideas and hints for making progress
ColorfulTiles: Ideas, Algorithm, Code

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

● "RGGBBBBRYYB" has solution 3, why?

● Indexing can help. Changing a cell's color?
  ▶ What about tile at $i-1$ and at $i+1$ (from $i$)
  ▶ Change cell 2? Change 3, 4, 5?
Two structures, two types of loop

- Loop by index or by element?
  - What do we do with each element, how to compare to element before? Or After?
  - Can do with either loop, much simpler with indexing loop!
Two structures, two types of a loop

● **What structure to use: string or list?**
  - After examining each element, operations?
  - At green, with green to left, change me to ____?
    • Use list to mark with 'x', advantage of four colors!
    • Remember strings are immutable
Solving UniqueNames APT

Bodhi

Penelope

Xerxes

Cayenne
Approach to solving APT

● Loop over classes, find name/year, repeat
  - Indexing loop? name/year, name/year, ...
  - Use range(x,y,2) for accessing every other entry

● Add all student names to a list or a set
  - How to make list contain no duplicates? Set
  - How to sort all the names? Sorted

● Combine ideas, and done!
Coding aspects, not algorithmic

- Given list of names...stored in taken ...
- Add a name to list or set, differences?
  - `taken.append(name)` or `taken.add(name)`
- Create a list of just the unique names?
  - `uniq = list(set(taken))`
- Create a sorted version of taken?
  - `st = sorted(taken)` -- list, string, or set?
- Create space-separated string with values?
  - `' '.join(joinable)` what type is joinable?
Challenge APT UniqueZoo

● Using sets and set operations can help
  ➢ Set intersection and set union
  ➢ Other set operations
● A | B, set union, A & B, set intersection
● B – A, set difference, B ^ A, symmetric diff
Nested Loops: FortunateNumbers

● How to loop over all characters in each string of a list, e.g., to count all occurrences?
  ➢ Looping over string in loop over list...
  ➢ Think only about inner loop first, ...

● Often useful to put inner loop in another function! Did this in Caesar.py

```python
for x in ['apple', 'pear', 'orange']:
    for ch in x:
        print ch
        values[alpha.index(ch)] += 1
```
Create "couples" from two lists A, B

● A name is fixed as the inner loop executes
  ➢ See output to reinforce this idea

A = ['sam', 'lou', 'chris']
B = ['terry', 'brook', 'val']
for aname in A:
    for bname in B:
        print aname,",",bname

sam , terry
sam , brook
sam , val
lou , terry
lou , brook
lou , val
chris , terry
chris , brook
chris , val
Answer Questions

Sets are unique and fast!

- **Consider importance of** \( \text{-- if } x \text{ in } Y : \)
  - How long does it take to find \( x \text{ in } Y \)?
  - What does it depend on? See Timings.py

- **How can we find if a book is in library?**
  - Assume all books shelved properly, then ...
  - If we can't assume that, what do we do?

- **Look up in set doesn't depend on set size!**
  - Cannot add list to set or set to set or ...

Compsci 101.2, Fall 2015