This week in Compsci 6/101

- Review sets and lists
  - What is a set, when is it useful
  - What operations available on sets

- New APTs, some of which leverage sets
  - Others use while loops, danger here?
  - What is a mastery APT?

- Hangman assignment: understanding modules
  - What we're doing this week, lead into cheating hangman

Programming Style

- Functions return a value
  - If they don’t they do anyway: None
  - Sometimes it's ok not to, the function has a side effect
  - Sometimes alter a parameter and return a value: side effect
  - Could be printing, could be altering global state
  - Could be ...

- Functions have a comment indicating their purpose
  - Describe functionality
  - Describe parameters
  - Describe return type
  - Describe exceptions (later)

How to build a program

- Create a module
  - Can be used by other modules via import
  - Collection of functions, later collection of classes
  - How do the functions in one module communicate?

- Sometimes functions in one module interact
  - Easy if one function calls another
  - Harder if state must be saved between function calls

- Enter global variables (not used in Hangman)
  - The scourge of human kind

Luis von Ahn (Duke 2000, Macarthur)

I am working to develop a new area of computer science that I call Human Computation. In particular, I build systems that combine the intelligence of humans and computers to solve large-scale problems that neither can solve alone.

An example of my work is reCAPTCHA, in which over 750 million people—more than 10% of humanity—have helped digitize books and newspapers.
Python Sets

- What do sets do (that lists and tuples cannot do?)
  - Very efficient membership queries, really efficient...
  - Can only store immutable elements, why?
  - Are iterable, but no particular order (what?)

- Set operators and methods interchangeable?
  - How do we compute set intersection
    - \( S \cap T \) v. \( S \text{.intersection}(T) \) v. \( S \text{.intersection_update}(T) \)
    - Compare \( S \&= T \)
  - Other operators: | (union) – (difference) ^ (symmetric diff)

APTs for practice: great minds ...

- Membercheck and SimpleWordGame
  - Set idioms
  - List comprehensions
  - Understandable code

- Why do we study more than one way to ...?
  - Expressive power
  - Neuronal excitement
  - Creating connections
  - We don’t have to

Set Operations from pictures

- [Venn Diagrams](http://en.wikipedia.org/wiki/File:Venn0111.svg)

Playing (word) games

- What is hangman and how is it played?
  - Identify crucial features from programming standpoint
  - What should we do first?
  - How do we test it?

- Designing program to be changeable
  - How do we add a GUI for selecting letters?
  - How do we add a GUI for drawing hanged person?

- How do we move toward cheating?