Compsci 101: Running Python

- What does it mean to run a program?
  - What does clicking on app/program do?
  - How do you run/test APT code, other Python code
    - Where does program start to execute/run?

- Control flow in Python
  - Changing order in which Python statements execute
  - Loops and if statements
  - Essential for writing real programs
    - Everyone with high BMI in 101?

How to solve an APT

- Two very, very, very important steps
  1. How to solve the problem with Paper, Pencil, (Calculator)
  2. How to translate problem-solving to Python

- Both steps can be hard, vocabulary and language are initially a real barrier
  - The more experience you have with Python, the easier step 2 will get
  - The more you understand the idioms and power of the language the more you can let step 2 influence step 1

- Step 1 is key, without it you won’t get anywhere

APT Pancake

- How do you solve this problem?
  - First steps: are there simple cases that can be solved immediately?
    - What are these for the pancake problem?
    - How will you identify with Python?
  - Sometimes it helps to know if you are on track, use Python to check your paper and pencil work

- Get specific, solve for 5, not N
  - Fix one parameter, vary the other
  - Identify the cases and continue

Three pancakes in a two-cake pan…

- Number of cakes in the system
  - First 5 minutes

- Number of cakes in the system
  - Second 5 minutes
Three pancakes in a two-cake pan...

- Number of cakes in the system
  - Third 5 minutes
- How many minutes to cook all three pancakes?

How to teach pancake flipping

- [http://www.youtube.com/watch?v=W_gxLKSsSIE](http://www.youtube.com/watch?v=W_gxLKSsSIE)
  - For longer, more complex robotic tasks
    - [http://www.youtube.com/watch?v=4usoE981e7I](http://www.youtube.com/watch?v=4usoE981e7I)

- Back to specifics:
  - Capacity = 5
  - Numcakes = 1, 2, ..., 5?
  - Numcakes = 6, 7, 8, 9, 10?
  - Numcakes = 11, 12, 13, 14, 15?
  - Is five special? 4? 3? 2?

Eclipse Interlude

- Finishing the Pancake problem
  - Translating problem-solving ideas to code
  - Control with if/elif: arithmetic with / and %

Lessons: special cases, abstractions

- There are special cases in many, many problems
  - Identifying them is important
  - Abstractioning them away when possible is important
    - Example: Pancake APT
      - What happens when everything fits in the pan?
      - Can there be a pan with no capacity?

- Translating ideas into (Python) code
  - How do we create interesting “heads”, “totem poles”?
  - How do create software for identikit?
  - How do we create Facebook, Foursquare, ...
How do you solve a problem like …?

- Translating English to Piglatin
  - Why is this fascinating?
  - Is this like translating English to German?
  - Is it like translating Python to bytecode?

- “downplay their unique quiet strength”
  - “ownplay-day eir-thay unique-way iet-quay ength-stray”
  - What are the rules for pig-latin?

---

Three versions of is_vowel

```python
def is_vowel(ch):
    if ch == 'e':
        return True
    if ch == 'a':
        return True
    if ch == 'i':
        return True
    if ch == 'o':
        return True
    if ch == 'u':
        return True
    return False
```

```python
def is_vowel(ch):
    c = "aeiou".count(ch)
    if c > 0:
        return True
    else
        return False
```

```python
def is_vowel(ch):
    return "aeiou".count(ch) > 0
```

---

Grace Murray Hopper (1906-1992)

- “third programmer on world's first large-scale digital computer”
  - US Navy: Admiral
  - “It's better to show that something can be done and apologize for not asking permission, than to try to persuade the powers that be at the beginning”

- ACM Hopper award given for contributions before 35
  - 2004: Jennifer Rexford
  - 2010: Craig Gentry: [http://www.youtube.com/watch?v=qe-zmHoPW30](http://www.youtube.com/watch?v=qe-zmHoPW30)
  - 2011: Luis von Ahn