Introduction to CS111
Part 2: Big Ideas

What is Computer Science?

- It’s not really about computers.
- It’s not really a science.
- It’s about imperative (“how to”) knowledge as opposed to declarative (“what is”) knowledge.
- Imperative knowledge is expressed via algorithms = computational recipes.
- “A computer language … is a novel formal medium for expressing ideas about methodology, not just a way to get a computer to perform operations. Programs are written for people to read, and only incidentally for machines to execute.”
  -- Harold Abelson and Gerald J. Sussman

Four big ideas

- Four important concepts are at the core of this course:
  1. Abstraction;
  2. Modularity;
  3. Divide, Conquer and Glue;
  4. Models
- These interrelated ideas are important in almost every discipline, but they’re at the core of CS.
- We will illustrate these ideas in several ways.
- Our goal is to help you think about problem solving in new ways.

Big Idea #1: Abstraction

User / Client

Implementer / Designer
Contract / API
Big Idea #2: Modularity

- Large systems are built from components called modules.
- The interfaces between modules are designed so they can be put together in a mix-and-match way.
- In computer programming, the goal is to design packages for maximum reusability.

Big Idea #3: Divide, conquer & glue

Divide
- problem P into subproblems.

Conquer
- each of the subproblems.

Glue (combine)
- the solutions to the subproblems into a solution S for P.

Big Idea #4: Models

- Need simple models to understand complex artifacts and behaviors.
- We'll draw lots of diagrams to predict what programs will do.

What will you build with these ideas?

We'll start with numerical calculations and graphics

- What is your name? Valentine
- How many classes are you taking this semester? 5
- How is the average time in class per week this semester? 2.5
- How many hours per week do you spend on extracurricular activities (including jobs)? 15
- How many hours per day do you sleep on average? 8
- Weekly time profile for Valentine:
  - 37.5 class hours: 0
  - 15.0 extracurricular hours: 0
  - 59.5 free hours: 0
  - 56.0 sleep hours: 0

Happy Friday Image: Lovely Day
Functions capture common patterns

**Divide/Conquer/Glue with Quilts**

**Conditionals**

```python
In [1]: isPalindrome('No lemon or melon')
Out[1]: False
In [2]: isPalindrome('No lemon, no melon')
Out[2]: True
In [3]: isOddAndPal('No lemon or melon')
Out[3]: True
In [4]: isOddAndPal('No lemons or melons')
Out[4]: False
In [5]: classify('No lemon or melon')  # Just a palindrome
   just a palindrome: No lemon or melon
In [6]: classify('No lemon, no melon')
   a palindrome: No lemon, no melon
In [7]: classify('No lemons, no melons')
   not a palindrome: No lemons, no melons
In [8]: classify('No lemons, no melons')
   neither a palindrome nor a palindrome: No lemons, no melons
```

**Iteration/Loops**

```python
In [8]: scrabbleScore('juxtapose')
Out[8]: 25
In [9]: scrabbleScore('quiz')
Out[9]: 22
In [10]: scrabbleScore('Wellesley')
Out[10]: 15
```
Data Structures: Lists and Dictionaries

In[16]: from wellesleyfrench import *
In[15]: getWeather(getWellesleyWeather(lat), 'Thursday', 'Rates')

Deli Egg Salad
Deli Soup - Fish Chowder
Breakfast Scrambled Eggs Or Waffles Veg
Homestyle Lunch - Hamburger Salad
Homestyle Dinner - Green Beans
Deli Turkey Roll
Homestyle Dinner - Steamed Corn
Breakfast Hardboiled Eggs Veg
Fusion - Turkey Ala King
Homestyle Dinner - Beef Ribs
Breakfast - Waffle Station Veg
Homestyle Dinner - Crumpled Spinach
Breakfast - Steel Cut Oatmeal
Global Grill - Grilled Kollusco
Fusion - Vegetarian Ala King
Homestyle Lunch - Baked Chicken Bruschetta
Pasta - Pasta
Homestyle Lunch - Garlic Toast
Homestyle Dinner - Baked Potato Bar
Global Grill - Potato Perogies
Pasta - Hard pasta
Deli - Cheese Veg
Global Grill - Potato Fancakes

Recursion

def power(x, n):
    if n == 0:
        return 1
    else:
        return x * power(x, n-1)

power(2, 3)  # Returns 8
power(2, 4)  # Returns 16

Processing Data from Web APIs

from last.fm import similar_artists

collection = similar_artists('Coldplay')
collection.title

collection.artists

collection.albums

Searching last.fm for Similar Artists

Your search party "Coldplay" found these results for similar artists:

The Colourist
Walk the Moon
Shepard
MoonWaves
Pooles

Smallpox
Tan La
Lights
American Authors
Foster the People

deepsix.com
On to Python! Unlearn what you have learned

I don’t think that word means what you think it means.
Python Intro Overview

- **Values:**
  - 10 (integer),
  - 3.1415 (decimal number or float),
  - 'wellesley' (text or string)

- **Types:** numbers and text: `int`, `float`, `str`

  ```
  type(10)  # Type of integer
  type('wellesley')  # Type of string
  ```

- **Expressions:** (they always produce a value as a result)

  ```
  len('abc') * 'abc' + 'def'
  ```

- **Built-in functions:** `max`, `min`, `len`, `int`, `float`, `str`, `round`, `print`, `raw_input`

---

The First Model: Variable as a Box

- Variables are names we make up (but, there are rules for creating these names)
- A variable name should appear for the first time in an **assignment statement**.

  ```python
  fav = 17  # assign
  fav + 3  # lookup
  # lookup and reassign
  fav = fav + 3
  ```

  - A value is stored in a “box”.
  - The variable “labels” the box.
  - When a variable is used in expressions, we lookup for the “box” with that name and read its value.
  - We can reassign a (new) value to a box.
  - If we use a name in an expression without using it in an assignment first, we get a **NameError**.

---

Example with cs1graphics

- **Version annotated to display coordinates for reference points.**

```python
from cs1graphics import *

paper = Canvas(400, 550, 'yellow')
head = Circle(50, Point(200,150))
paper.add(head)

# The torso
torso = Rectangle(100, 200, Point(200, 300))
torso.setFillColor('gray')
paper.add(torso)

message = Text('If I only had a heart!', 20)
paper.add(message)
message.setFontColor('red')
message.moveTo(200, 25)
```

**cs1graphics – A library for graphics**

- Defines names used in this program: `Canvas`, `Rectangle`, `Text`, etc.
- Assignment statement that:
  - Creates object, assigns it to a variable
  - Displays an empty canvas
  - Assignment statement (`value + name`)
  - Statement: method call to display circle object on canvas window
  - Statement: method call to change object’s internal state, also visible on the canvas.

- **SOME CODE OMITTED**

- Statement: method call to change reference point and display object in new position.
Expressions vs. Statements

They always produce a value:

10
10 * 20 - 100/25
max(10, 20)
int("100") + 200
fav
fav + 3
"pie" + " in the sky"

They perform an action (that can be visible, invisible, or both):

print(10)
age = 19
paper = Canvas(400, 550, 'yellow')
paper.add(head)

Statements may contain expressions, which are evaluated before the action is performed.

print('She is ' + str(age) + ' years old.')

Expressions are composed of values, operators, variables, functions, and any combination of 0 or more of them.