61A Lecture 1

Friday, August 30, 2013
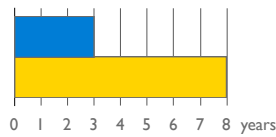
Welcome to Berkeley Computer Science!

The Course Staff

John DeNero

Google
Cal

0 1 2 3 4 5 6 7 8 years
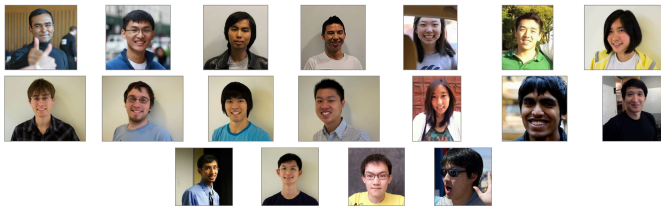
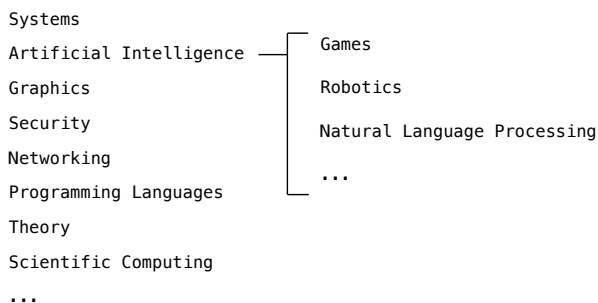**TAs** hold discussion sections, labs, and *office hours*

Readers are your personal programming mentors
Lab Assistants ensure that you don't get stuck for too long

An Introduction to Computer Science

What is Computer Science?

The study of   What problems can be solved using computation,
How to solve those problems, and
What design choices lead to effective solutions.

Systems
Artificial Intelligence ——— Games
Graphics   Robotics
Security
Networking   Natural Language Processing
Programming Languages ——— ...
Theory
Scientific Computing
...

What is This Course About?

- A course about managing complexity

  - Mastering abstraction

  - Programming paradigms

  - Not about 1's and 0's

- An introduction to Python

  - All the features we really need: introduced today

  - Understanding through implementation

  - How computers interpret programming languages

- A challenging course that will demand **a lot** of you

What is This Course About?

Plone Conference. Photo courtesy of Kriszta Szita

Course Logistics and Policies

Alternatives to This Course

CS 61AS: Self-paced 61A

CS 10: The Beauty and Joy of Computing

Course Policies

The purpose of this course is to help you learn

The staff is here to make you successful

All the details are online:

http://inst.eecs.berkeley.edu/~cs61A/fa13/about.html

Collaboration

- Discuss everything with each other
- **EPA:** Effort, participation, and altruism
- Homework can be completed with a partner
- Projects *should* be completed with a partner
- Find a project partner in your section (if you can)

The limits of collaboration

- One simple rule: Don't share your code, except with partners
- Copying project solutions is a serious offense!
- We really do catch people who violate the rules
  - We also know how to search the web for solutions
  - We let computers detect copying for us

Expressions

Types of expressions

An expression
describes a computation
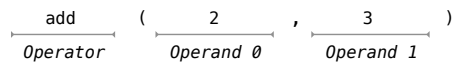and evaluates to a value

$18 + 69$

$\dfrac{6}{23}$

$\sin \pi$

$f(x)$

$\sqrt{3493161}$

$\displaystyle\sum_{i=1}^{100} i$

$|-1869|$

$\dbinom{69}{18}$

Call Expressions in Python

All expressions can use function call notation

(Demo)

Anatomy of a Call Expression

```
add       (     2      ,      3      )
Operator      Operand 0      Operand 1
```

Operators and operands are expressions

So they evaluate to values

Evaluation procedure for call expressions:

1. Evaluate the operator and operand subexpressions

2. Apply the function that is the value of the operator subexpression to the arguments that are the values of the operand subexpression

Evaluating Nested Expressions

Evaluating Nested Expressions

Operand 0 "subexpression"
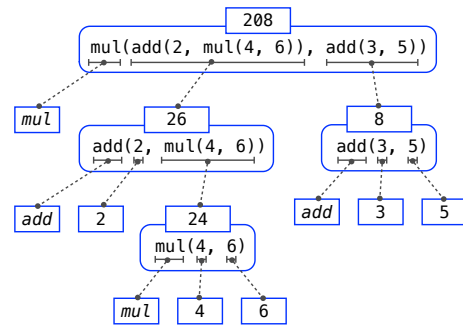
Expression tree
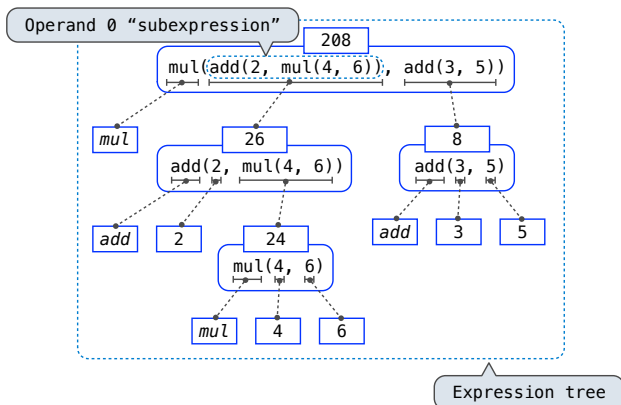
Data, Functions, and Interpreters

Data: The things that programs fiddle with

"The Art of Computer Programming"

2

Shakespeare's 37 plays

Donald Knuth
(Ka–NOOTH)

Functions: Rules for manipulating data

Count the words in a line of text

Add up numbers

Pronounce someone's name

Interpreter: An implementation of the procedure for evaluation