61A Lecture 1

Wednesday, August 24, 2016

Welcome to CS 61A!

I'm John DeNero

How to contact John:

denero@berkeley.edu

piazza.com/berkeley/fall2016/cs61a

John's office hours:

781 Soda

Monday & Wednesday 11am – 12pm

By appointment: denero.org/meet

The 61A Community

45 undergraduate student instructors / teaching assistants (TAs):

- Teach lab & discussion sections
- Hold office hours
- Lots of other stuff: develop assignments, grade exams, etc.

45+ tutors & mentors:

- Teach mentoring sections
- Hold office hours
- Lots of other stuff: homework parties, mastery sections, etc.

200+ lab assistants help answer your individual questions

1,500+ fellow students make CS 61A unique

Parts of the Course

Lecture: Videos posted to cs61a.org before each live lecture

Lab section: The most important part of this course (*next week*)

Discussion section: The most important part of this course (*this week*)

Staff office hours: The most important part of this course (*next week*)

Online textbook: http://composingprograms.com

Weekly homework assignments, three exams, & four programming projects

Lots of optional special events to help you complete all this work

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 techniques lead to effective solutions

- Systems
- Artificial Intelligence
  - Decision Making
  - Robotics
  - Natural Language Processing
    - Answering Questions
    - Translation
    - ...
  - ...
- Graphics
- Security
- Networking
- Programming Languages
- Theory
- Scientific Computing
- ...

What is This Course About?

A course about managing complexity

- Mastering abstraction
- Programming paradigms

An introduction to programming

- Full understanding of Python fundamentals
- Combining multiple ideas in large projects
- How computers interpret programming languages

Different types of languages: Scheme & SQL

A challenging course that will demand a lot of you

Alternatives to CS 61A

CS 10: The Beauty and Joy of Computing

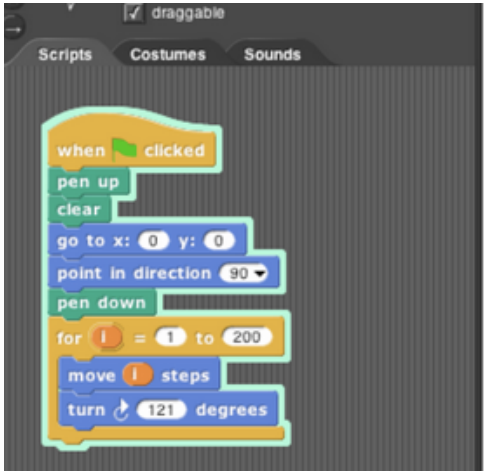
Designed for students without prior experience

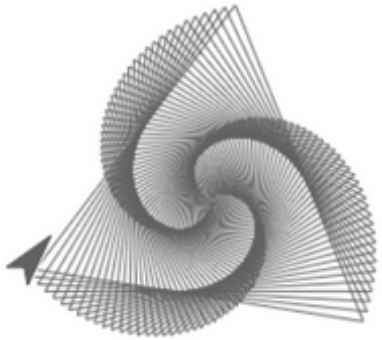
A programming environment created by Berkeley, now used in courses around the world and online

An introduction to fundamentals (& Python) that sets students up for success in CS 61A

Taught in Fall 2016 by Dan Garcia

More info: cs10.org





Data Science 8: Foundations of Data Science

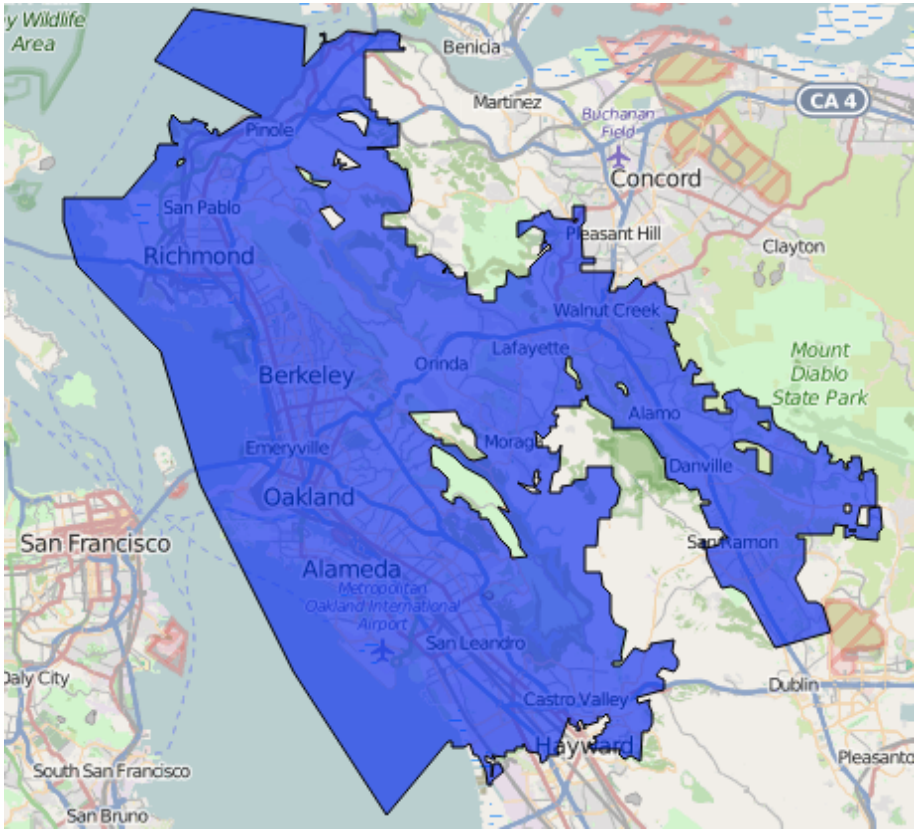
Fundamentals of computing, statistical inference, & machine learning applied to real-world data sets

Great programming practice for CS 61A

Cross-listed as CS C8, Stat C8, & Info C8

Taught in Fall 2016 by Ani Adhikari

More info: data8.org & databears.berkeley.edu



Course Policies

Learning

Community

Course Staff

Details...

http://cs61a.org/articles/about.html

Collaboration

Asking questions is highly encouraged

- Discuss everything with each other; learn from your fellow students!
- Homework can be completed with a partner
- Projects should be completed with a partner
- Choose a partner from your discussion section

The limits of collaboration

- One simple rule: Don't share your code, except with your partner
- Copying project solutions causes people to fail
- We really do catch people who violate the rules, because...
  - We also know how to search the web for solutions
  - We use computers to check your work

Build good habits now

Expressions

Types of expressions

An expression describes a computation and evaluates to a value

$18 + 69$

$\frac{6}{23}$

$\sin \pi$

$\log_2 1024$

2^{100}

$f(x)$

$\sqrt{3493161}$

$7 \bmod 2$

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

$\lim_{x \to \infty} \frac{1}{x}$

$|-1869|$

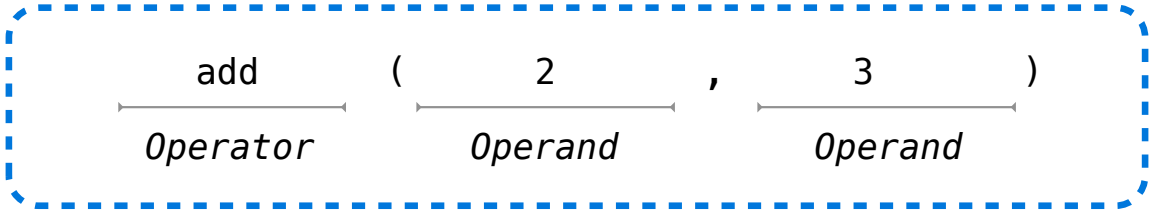
$\binom{69}{18}$

Call Expressions in Python

All expressions can use function call notation

(Demo)

Anatomy of a Call Expression



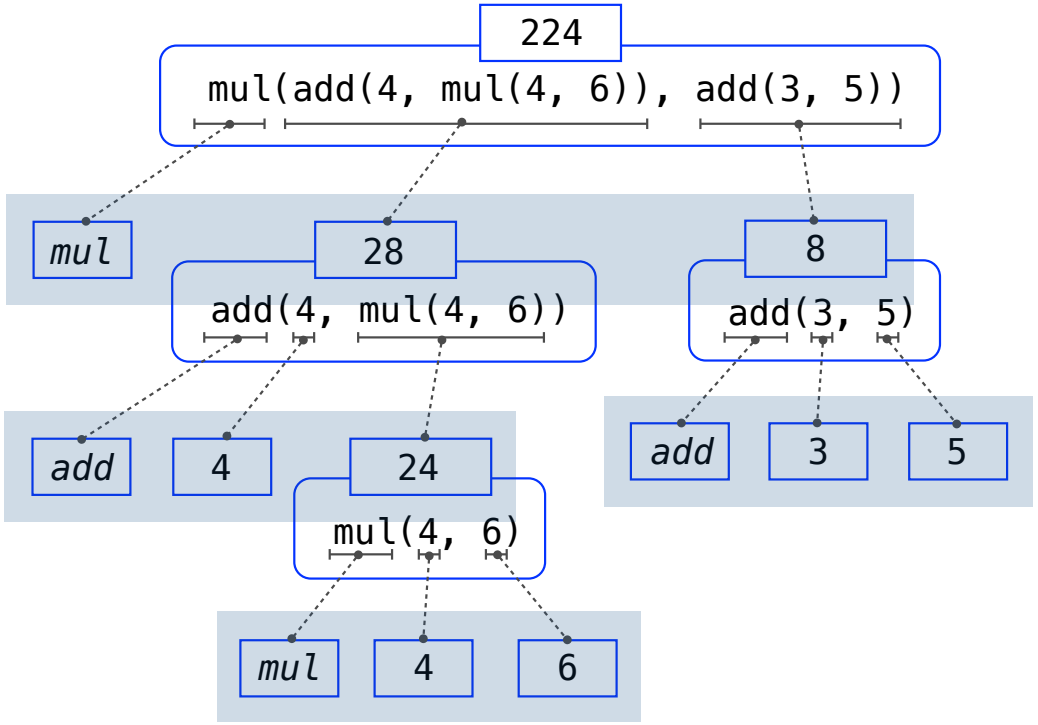
Operators and operands are also expressions

So they evaluate to values

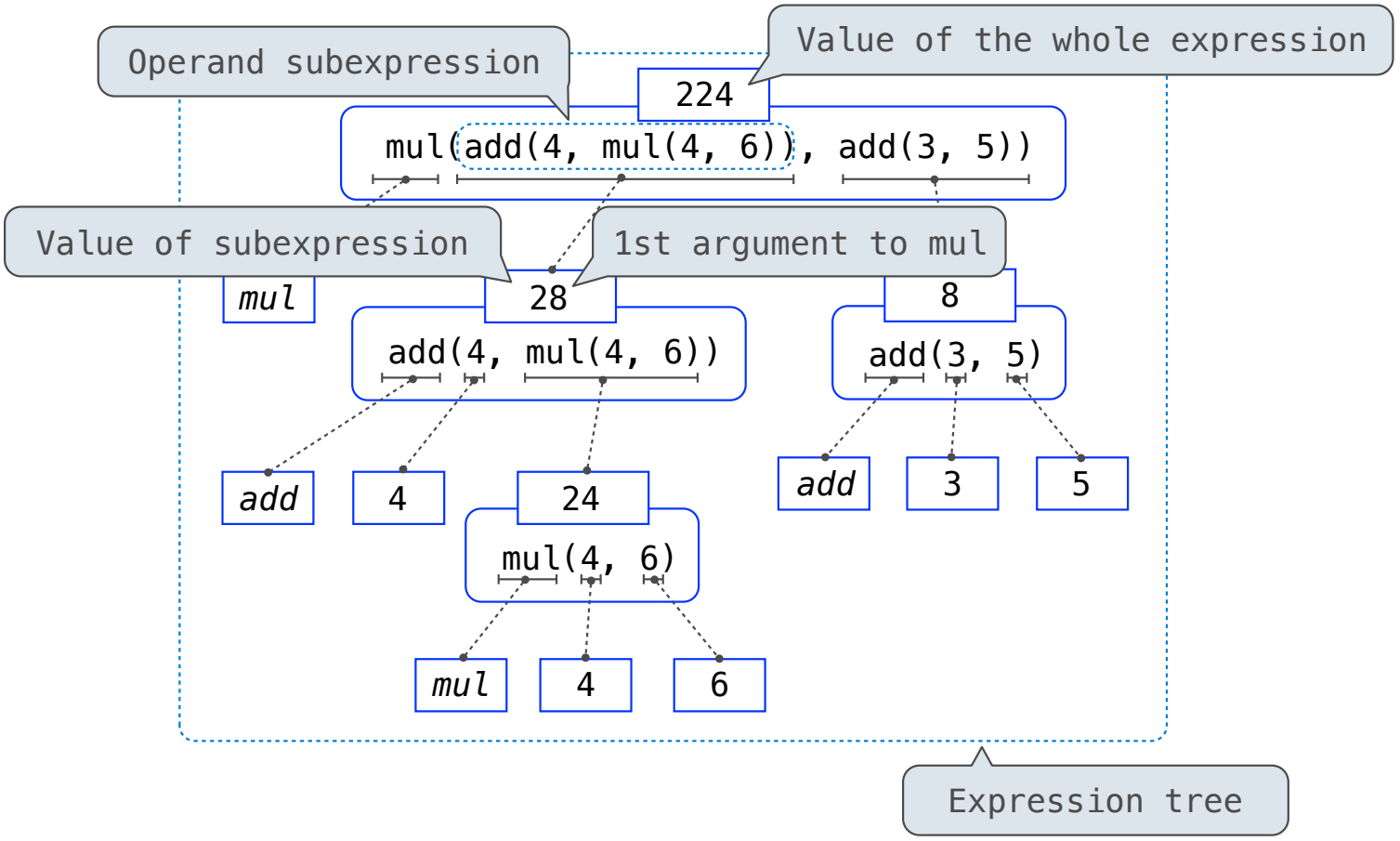
Evaluation procedure for call expressions:

1. Evaluate the operator and then the 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



Functions, Objects, and Interpreters

(Demo)