6.009: Fundamentals of Programming

Lecture 0: Environment Diagrams

Adam Hartz

hz@mit.edu

31 January 2022

6.009: Goals

Our goals involve helping you develop your programming skills, in multiple aspects:

- **Programming:** analyzing problems, developing plans
- **Coding:** translating plans into Python
- **Debugging:** developing test cases, verifying correctness, finding and fixing errors

6.009: Goals

Our goals involve helping you develop your programming skills, in multiple aspects:

- **Programming:** analyzing problems, developing plans
- **Coding:** translating plans into Python
- **Debugging:** developing test cases, verifying correctness, finding and fixing errors

So we will spend time discussing:

- high-level design strategies
- ways to manage complexity
- details and "goodies" of Python
- a mental model of Python's operation
- testing and debugging strategies

6.009: Goals

Our goals involve helping you develop your programming skills, in multiple aspects:

- **Programming:** analyzing problems, developing plans
- **Coding:** translating plans into Python
- **Debugging:** developing test cases, verifying correctness, finding and fixing errors

So we will spend time discussing:

- high-level design strategies
- ways to manage complexity
- details and "goodies" of Python
- a mental model of Python's operation
- testing and debugging strategies

...but discussion only goes so far.

6.009: Pedagogy

Learning to program is a lot like learning a musical instrument or a sport. How does one learn those things?

6.009: Pedagogy

Learning to program is a lot like learning a musical instrument or a sport. How does one learn those things?

Just like music/sports, practice is key!

To improve as a programmer, you have to program.

And 6.009 asks you to program...a lot!

6.009: Pedagogy

Learning to program is a lot like learning a musical instrument or a sport. How does one learn those things?

Just like music/sports, practice is key!

To improve as a programmer, you have to program.

And 6.009 asks you to program...a lot!

- Labs give opportunities to practice new techniques/skills to solve interesting problems.
- Lectures/recitations equip you with tools useful for attacking those problems.
- Checkoffs and office hours give opportunities to receive expert feedback.

6.009: A Typical Week

A typical week centers around a lab assignment, supplemented by instructor presentations and with lots of help available.

- **Lecture:** Monday, 11am-12:30pm in 26-100
- **Recitation:** Wed, 1-hour blocks from 9am-4pm

- **Open Lab Hours**
  - Mon-Thu Evenings, 7pm-10pm, room 34-501
  - Additional times TBA

Labs: the Heart of 6.009

Logistics:

- Typically issued Friday mornings
- Mix of conceptual questions and writing code (Python 3.6+, 3.10 recommended)
- Sometimes, some questions are due before lecture or before recitation
- Bulk of the lab is due the following Friday at 5pm Eastern
- Checkoff meetings are due on Wednesday at 10pm Eastern

Cool Problems!

- Audio/Image Processing, Games, LISP Interpreter, ...

Why Are You Here?

What are your goals for your time at MIT?

For your time in 6.009?

Getting the Most Out of 6.009

Lectures/Recitations:

- Step 1: Come to lecture/recitation, and participate!
- Take notes *in your own words* and review them later
- **Ask questions!** We want to have a conversation.

# Getting the Most Out of 6.009

Lectures/Recitations:

- Step 1: Come to lecture/recitation, and participate!
- Take notes *in your own words* and review them later
- **Ask questions!** We want to have a conversation.

Labs:

- Start early (labs are week-long assignments)
- Formulate a plan before writing code
    - Try to understand the problem thoroughly before writing code
    - When things go wrong, step away from the code and revisit the plan
- Work through problems on your own
- Ask for help when you need it!
    - Labs are intentionally challenging
    - Bugs are a natural part of life
    - Lots of opportunities for help (office hours / forum)
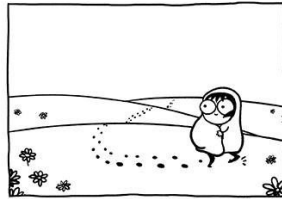
© Sarah Andersen

Academic Integrity

Our goal is that *every student* develops these skills throughout the course.

Collaborating too closely with others (our outsourcing pieces to other students or the web) can rob you of an opportunity to develop those skills in yourself.

Please read our policies about academic integrity carefully.

https://py.mit.edu/spring22/info/academic_integrity

6.009: Web Site

Just about everything in 6.009 happens via the web site:

<div align="center">

`http://mit.edu/6.009`

or

`https://py.mit.edu`

or

`https://6009.rocks`

</div>

Check Yourself!

What happens when the following program is run?

```python
functions = []
for i in range(5):
    def func(x):
        return x + i
    functions.append(func)

for f in functions:
    print(f(12))
```

1. It prints 12, then 13, then ..., then 16
2. It prints 13, then 14, then ..., then 17
3. It prints 16, then 15, then ..., then 12
4. It prints 17, then 16, then ..., then 13
5. A Python error occurs
6. Something else