

CS 150: Intro to Computer Science

Lecture 1: Introduction

Please pick up a syllabus and student info sheet in the back row, and start filling out the info sheet.

Sign in to the waitlist form if you are on/would like to be on the waitlist.

Cynthia Taylor

Oberlin College

February 3rd, 2014

Instructor: Cynthia Taylor

- Research: Efficiently and transparently distributing device drivers over networked systems
- Please call me either Cynthia or Professor Taylor
- <http://occs.oberlin.edu/~ctaylor>

Office Hours

- Wednesday, 11am – 12pm
- Friday, 3:30pm – 5pm
- By appointment
- King 229

Resources

- Class web page (for schedule, readings, assignments, grading policy, etc)
 - <http://www.cs.oberlin.edu/~ctaylor/150SP14/>
- Blackboard (for grades, forum, pdfs of lecture)
 - <https://blackboard.oberlin.edu/>

Email Policy

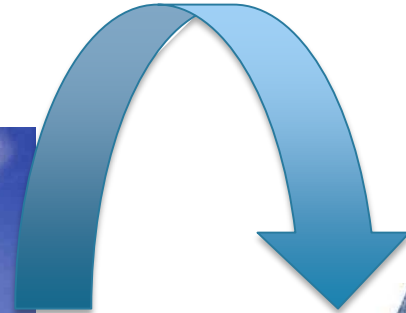
- Please use forum rather than email
 - Your classmates benefit from your questions
 - Your classmates can answer your questions
 - I will check the forum frequently
- I will attempt to respond to email within 24 hours

Clickers!

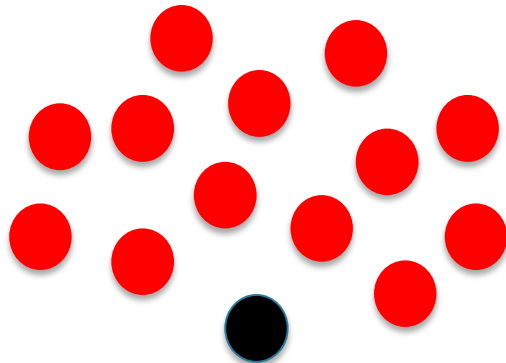


- Lets you vote on multiple choice questions in real time.
- Like pub trivia, except the subject is always computer science.

Why do we have lecture?

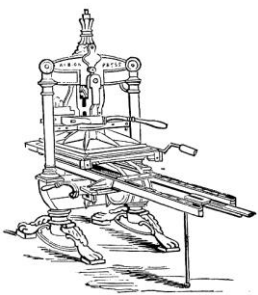


Why do we have lecture?



Why do we have lecture?

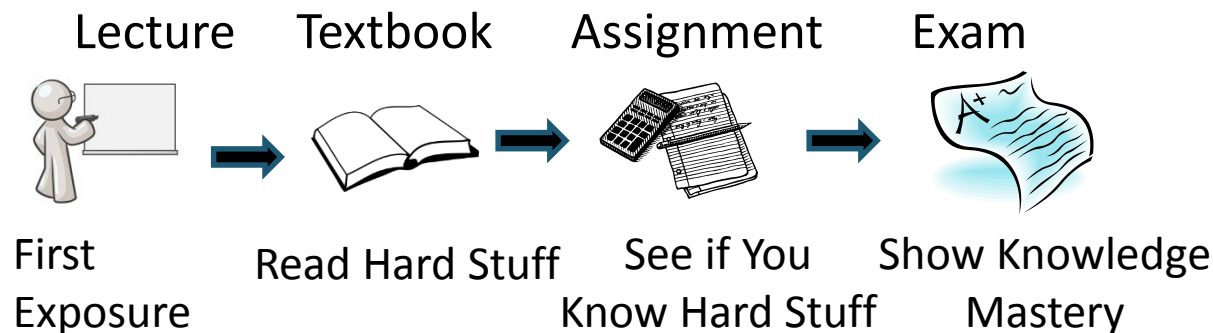




GREAT Innovations:

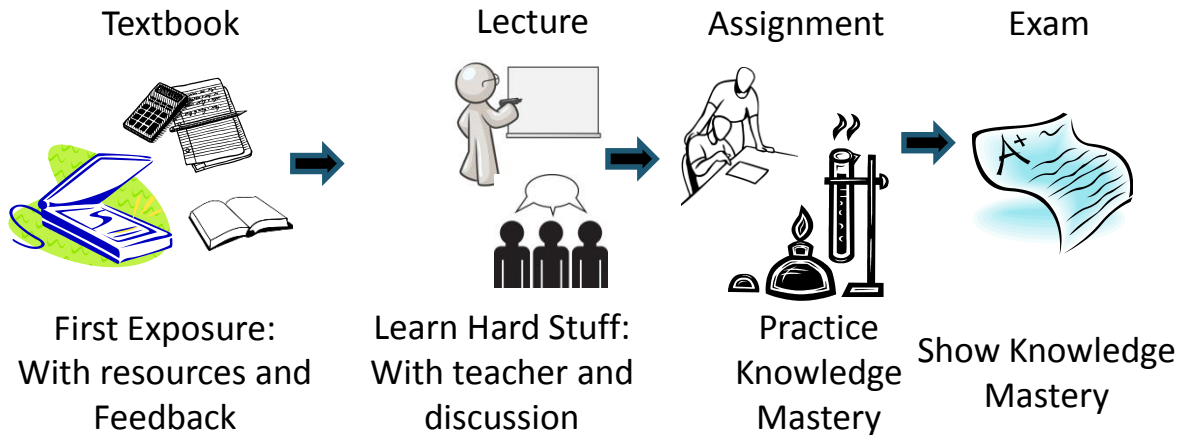
The printing press, The web

- You don't have the trust the monk!
 - Read it and analyze for YOURSELF!
 - If I rephrase it for you, what purpose does that serve?
- Traditional class structures often look like:



- You get very little opportunity for “expert” **feedback**

Peer Instruction-Based Design



- Greater opportunity for expert **feedback!**
- Research on how people learn:
 - Everyone constructs their own understanding
 - I can't dump understanding into your brain
 - To learn, YOU must actively work with a problem and construct your own understanding of it

Lecture: Peer Instruction

- Pose carefully designed question
 - Solo vote: Think for yourself and select answer
 - Discuss: Analyze problem in teams of 3
 - Practice analyzing, talking about challenging concepts
 - Reach consensus
 - If you have questions, raise your hand and I will come over
 - Group vote: Everyone in group votes
 - You must all vote the same to get your point
 - Class wide discussion:
 - Led by YOU (students) – tell us what you talked about in discussion that everyone should know!

Why Peer Instruction?

- You get to make sure you are following the lecture.
- I get feedback as to what you understand.
- It's less boring!
- Research shows it promotes more learning than standard lecture.

Example: The best dinosaur is:



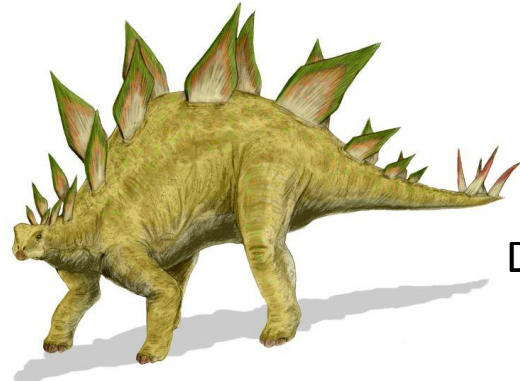
A. T-rex



B. Raptor



C. Triceratops



D. Stegosaurus

E. Some other dinosaur

Assigned Groups

- Three people each.
- I will assign groups for next class.

Does this mean I have to show up to class?

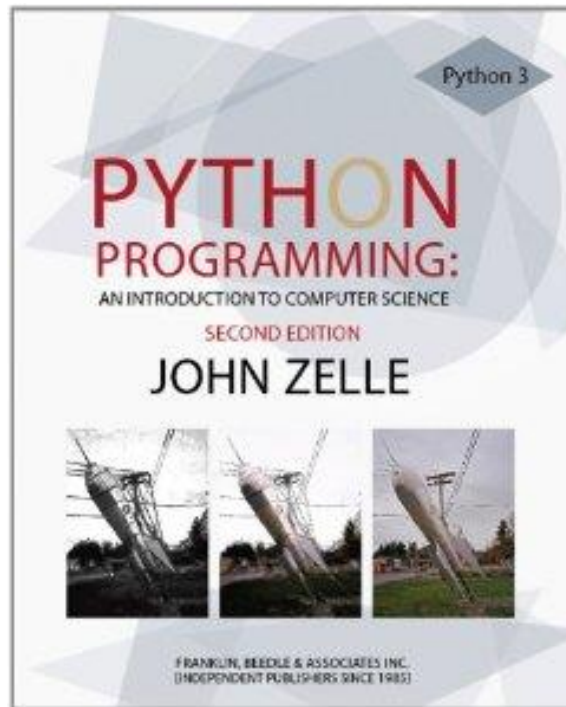
- Yes
- I will drop your lowest 3 class participation scores

Register your clicker online so I can give you points

- <http://www1.iclicker.com/register-an-iclicker/>
- Your votes will be saved before you register, just not associated with you
- Your “student id” should be what you use to log into blackboard, usually first initial, last name.
 - Mine is ctaylor

2 weeks

Textbook



- *Python Programming: An Introduction to Computer Science, by John Zelle (2nd edition)*
- You MUST own this book

Grading

- Weekly labs (40%)
- Weekly pre-labs (5%)
- Two in-class tests (25%)
- Final exam (20%)
- Class participation and lab attendance (10%)

Late Policy

- No late pre-labs are accepted.
- You get 2 late days for labs, which will be automatically applied.
- After your late days are gone, labs lose 50% per day
- No labs accepted more than 2 days late

Collaboration Policy

- Please do
 - Post on the forum
 - Talk with classmates about assignments
- Please don't
 - Copy someone's code that you don't understand

How to get an A in this class

- Come to class. Talk in class.
- Go to lab. Do the labs.
- Do the prelabs.
- If you don't understand something, come to office hours. Or lab helper hours. Or get a tutor.

Questions?

- All of this info on class website & blackboard (including these slides!)

Please fill out the intro sheet

What is Computer Science?

making algs for
computers to solve
problems

make comps do stuff

What is an algorithm?

process data jobs through
sorting
recipe
knitting pattern

What is an algorithm?

- Finite number of instructions or steps
- Each instruction is well-defined (not ambiguous)
- Eventually halts
- Solves a general class of problems

Python



- High level (more abstract) language
- Designed to be highly readable
- Designed to be fun!
 - named after Monty Python

Python as a Calculator

- The easiest way to experiment with Python is to enter statements into the Python shell
- Python supports many common mathematical operators
- Operators include + (addition), - (subtraction), * (multiplication), ** (exponentiation), / (division), % (remainder)
 - e.g. `11+56` at the shell gives `67`

Hello World

Reading

- Next lecture: Variables, Expressions and Assignments
 - Section 2.1-2.5
- Please let me know about:
 - Your preferred name, if different than roster name
 - Athletic schedules
 - Disability accommodations