



The challenge of teaching 10^N passionate students ($N=0,1,2,3,4\dots$)

The psychology, art, and science of seriously playful computer science education

Lawrence Angrave, University of Illinois at Urbana-Champaign

CS @ Illinois: Introduction to Computer Science

> Big enrollment challenge:

800 students per year - we're at capacity
~10% of u-grad campus population take this course
Other, larger, gentler intro CS courses for non-majors

200 students per lecture
40 students per lab section

> Innovations & lessons-learned '06-'12

Student passion & interest is key. Layer & spiral content. Expect students to add depth outside of lecture. Lecture time is to present patterns of analysis, introduce concepts, solve problems not present numerous facts.

Engaging lectures: No slides. Use a paper handout to work on mini-problems. Teach patterns of problem-solving using broken & incomplete examples.

Physiological warfare: Understand students' needs, backgrounds, abilities, distractions and interests.

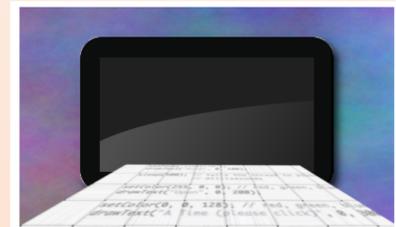
> Checklist for effective teaching

- ✓ Do students *feel* empowered by the material or do they believe course content only has short-term "exam-value"?
- ✓ Lectures are a "Collaborative, engaging performance"
- ✓ Create rapid feedback mechanisms
- ✓ Identify student-led exploration opportunities
- ✓ Use minimally specified team-based open-ended projects *plus* tightly-specified programming assignments *plus* "one minute" online exercises.
- ✓ Minimize student content search time, "1 click-away-to-learn: assignments contain relevant materials, examples etc.
- ✓ 40 hours/week of face-to-face and virtual office hours.
- ✓ 50 assistants: unpaid; A+ students from prev. semesters
- ✓ Use honors students as one-on-one mentors
- ✓ Active discussion forum (we use Piazza)
- ✓ Regrade opportunities
- ✓ TA and undergrad assistant training
- ✓ Explicit learning objectives
- ✓ Cohesive experience: Assignments are prep. for exams
- ✓ Course has a reputation for catching cheaters
- ✓ Ask, "Do students complete my lecture / course with more passion and interest for <X> than when they started?"

Coursera Massive Open Online Course

"Creative, Serious and Playful Science of Android Apps" (2013)

<https://www.coursera.org/course/androidapps101>



> Hypothesis & manifesto

"Empowering students outweighs the benefits of sandboxed environments. Students quickly tire of isolated sandboxes, whereas passionate students, using crafted course materials, eagerly rise to the challenge of real tools and environments, no matter how complex and difficult they may first appear."

> Huge enrollment challenge

31,000+ students
Diversity of skills, abilities, maturity
Varied student computing resources OSX XP Win7
- We provide a pre-configured virtual machine.

> Content & testing challenges

Online videos with embedded questions
Include interactive web pages in assignments

Break traditional long textbook format into short conversational-style Q&A with hyperlinks. Provide a fun, constrained exploratory navigation aka "ChooseYourOwnLearning Adventure"
- not just a big-list-o-links, or "click next"
Avoid "fact dumping" on novices
this is hard - we want to write *everything* down!

Scaffolding: Use broken and incomplete projects to maximize affective experience
Content must also support weaker students - provide smaller steps
Assignments use automated unit tests for rapid feedback

> About the course

"This course is a novice-friendly and delightful introduction to computer science and programming Android-apps for smart-phones and tablets. No prior programming knowledge is necessary. In this course you'll have fun learning to create an app for modern Android devices such as the Nexus tablet. You'll use the programming tools that Android software developers use and build a complete and useful app during this course. Along the way, we'll introduce fundamental computer science principles and programming ideas that power today's smart-phone and tablet apps."