Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education

Dan Garcia, Ph.D.
Lecturer SDL, BECS Dept. UC Berkeley (on Sabbatical in Melbourne until 2008)
www.cs.berkeley.edu/~ddgarcia/

Overview

- Quickie background
- SIGCSE plug
  - This talk from 3 in 2007
  - ACE2008 in Wollongong
- Improving CS education
  - Teaching Tips
  - Best Practices from survey of US schools
- ACM Education Board

Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education
www.cs.berkeley.edu/~ddgarcia

Teaching Tips Archive

- Goal: Gather great CS teaching tips on a wiki
- 4 seasoned educators seeded with 10 tips each
- Categories: Lecturing, Office (hrs), Staff, Exams, Labs, Section, Projects, Grading, Meta

Dan Garcia
UC Berkeley
Owen Astrachan
Duke Univ
Nick Parleante
Stanford Univ
Stuart Reges
U Washington

Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education
www.cs.berkeley.edu/~ddgarcia

Share Your CS Passion (Lecture)

www.siggraph.org/publications/video-review/
ucbugg.berkeley.edu/

- For each of these, feel free to shout "Amen, Brother!", and why it did / didn't work for you
- I show SIGGRAPH animations before every lecture
- I talk about opportunities for students to join my graphics group (UCBUG)
- The videos are available for $40/video to ACM members

Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education
www.cs.berkeley.edu/~ddgarcia

Team-teach (Meta)

www.cs.berkeley.edu/resources/education/liveways.html

- “Classroom Observation: The Observer as Collaborator”
  - LuAnn Wilkinson

- Rather than having a single lecturer / TA cover every lecture / section, if two can attend each other's sections, pair up & alternate weeks.
- On week they do everything (TAs would cover both sections)
- TAs in the section
- TAs in the section
- TAs in the section
- TAs in the section
- TAs in the section

Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education
www.cs.berkeley.edu/~ddgarcia

Clickers & Peer Instruction (Lecture)

www.interwritelearning.com/products/prs/
www.einstruction.com

- We instituted them in our lower-div classes
- Cost: ~$40/clicker
- Two pricing models: subscription or purchase
- Midway through lecture, ask "concept test"
  - Students vote w/clicker
  - 2 min peer instruction
  - Must reach consensus
  - "Team" votes again
  - We discuss answers
  - Real-time feedback!
  - Can ask real-time Q!

Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education
www.cs.berkeley.edu/~ddgarcia
Empower TAs (Staff)

- Ask TAs teach a lecture
- Have TAs and readers help write, beta-test, grade exams
- Let TAs participate in syllabus discussions
- Open your life to your staff
- Share cell numbers (great for emergencies)
- Grade exam at your home
- Take your admins out to a fancy lunch every term

Bring Laptops to Exams (Exams)

- Visible timer
  - Intermittent vs polling!
- Show exam bug corrections on screen
  - As you can fix right there
- Auto-announce time left
  - echo 'say "You have two hours left!"' | /usr/bin/osascript
  - sleep 1800 &
  - echo 'say "You may interrupt"' | /usr/bin/osascript
  - plaympeg ./epilogue.mp3

A Survey of Intro CS Curricula

- Goal: Understand what top US PhD-granting schools did in their intro CS curricula
- Survey sent out, collected, results presented
  - Today I'll talk about the best practices we noticed

Schools (2006 “US News & WR”)
**What works really well and what is broken?**

- Brown
  - High TA-student ratio
  - Intro Experience Disparity
- Cal
  - Closed required labs
  - Undergrad teaching staff
  - More pair programming
- CalTech
  - CS1 is several mini-courses.
  - Each student only takes 1 of these a term (at their choice).
  - A course can be repeated three times for credit.
- CMU
  - All courses taught in small sections - no large lectures!

What has worked well for us is to reduce the sources of information and to direct students’ efforts towards more active and collaborative learning both in and out of class.

**What works well and what is broken?**

- U Penn
  - Compelling programming assignments
- USC
  - Intro experience disparity
- UT Austin
  - Students move well through CS5
  - Java doesn’t allow them to learn memory management!
  - One course is “peeked out” which could turn off non-programmers.
- UW
  - Procedures-first intro to OOP
- U Wisconsin
  - Information overload
  - Reduce sources of info
  - Active and collaborative learning!

```
information overload has been mentioned repeatedly to me by students as a principal problem in many of their courses.
It is not uncommon for courses today to have a textbook or two, course notes, as well as lecture slides which overwhelm students with presentations of the same or similar material.
Add to that course enroll, web pages/links, blogs, etc. and students get trapped spending too much time reading and reviewing and insufficient time actively applying their newly gained knowledge.
```

```
What has worked well for us is to reduce the sources of information and to direct students’ efforts to more active and collaborative learning both in and out of class.
```

Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education

www.cs.berkeley.edu/~ddgarcia 15/24

**What are the innovative techniques being deployed?**

- Georgia Tech
  - Computational Media degree!
  - Different entry points to major
  - CS1 for different majors
    - Robotics
    - Media Computation
  - Flexibility
  - CS Majors can take any CS!
  - Contextualization
    - CS1s are typically dry!
    - Students today use Flikr, youtube, etc -- leverage that!
    - Allow the data to be theirs!
      - Movies, sounds, etc.

**What are the innovative techniques being deployed?**

- MIT
  - Overhauling ECEs curriculum
    - Had been unchanged for nearly 30 years
    - 6.001 (Schemel out, Python in!)
    - 6.01 and 6.02 to be new mandatory intro courses
    - Offer intro to dept.
    - Cut broadly across ECES
    - Strong hands-on component
    - Build robots, apply concepts
      - Radio, Fourier transforms, networks, large systems
    - Then choose 3 or 4 subjects from foundation courses
      - E.g., Circuits, Signals, Systems

Teaching Tips, Best Practices, and Other Initiatives to Improve CS Education

www.cs.berkeley.edu/~ddgarcia 15/24
**Conclusion, pt II**

- There is so much more analysis I didn’t present!
  - It’s all online; I encourage you to see our results
- There is some great innovation in the US!
  - MIT, GaTech, UC Berkeley, Caltech, UIUC, ...
- It would be great to learn about the innovation going on here to take back with me!

Survey!

---

**ACM Education Activities**

- The ACM Education Board has three active task forces
  - Pipeline Crisis, Image
  - This was “enrollment”
  - Technology and Tools
  - Curriculum
- As a member of the Board, I want to learn what your problems are and how ACM could help!

---

**One Take on Enrollment**

- Our numbers roughly half that of the 2000 peak

UC Berkeley Computer Science Annual Enrollment, 2002-present

---

**Response: Image Site**

- Numbers across US half that of 2000
  - Some coming back
  - Need for computing professionals growing
- Computing image poor
  - Nerds, no life, just coding
- Computing Degrees & Careers site launched
  - Supplemental brochure sent to all US high schools

---

**Technology and Tools Site**

- Problem
  - Computing educators (newbies and vets) don’t know of all available classroom technology and tools (too hard to keep up)
- Solution (still in beta)
  - Build site for them
    - Just links w/comments
    - What is missing?
  - Support Web 2.0
    - Ratings, tags, how it worked for you, etc.

---

**Conclusion, pt III**

- Take-away messages
  - Teaching tips
    - What are yours?
    - Contribute to our wiki
  - Best practices
    - What are yours?
    - Read survey for more
  - ACM initiatives
    - How could ACM help?
    - Talk to me about issues!
- Let’s collaborate!