The Grace Hopper Celebration of Women in Computing 2009

Best Practices for Introductory Computer Science

Valerie Barr, Jessica Bayliss, Monisha Pulimood, Susan Rodger, Ursula Wolz
What’s Wrong?

- **Taulbee Survey:**
  - approximately 50% fewer students entered computer science (CS) in Fall of 2007 compared to Fall of 2000
  - 2007-2008 BS degree production down again
- **CS major has a 30-40% attrition rate**

- **Overall, education has problems**

12th Graders Views of School

What Can Help?

• Choosing the Right Tools

• Peers, Teams, Expertise, Ownership

• What makes a “good” Computer Science problem?

• Don’t Dumb it Down – But They Can!

Which of these things have we done and how have we done them at each of our schools?
• Peer Led Team Learning
  • Students solve problems related to course
  • In small groups of size 4 to 8
  • Led by Student peer leader
  • Non-majors course (Alice) and CS 1(Java)

• Benefits to Undergraduate Students
  • Grades, lower drop rates, social groups, understanding

• Benefits to Peer Leaders
  • Confidence, leadership, better understanding material
RIT: Game Software Development 1-3

- Initially a MS funded grant to bring context into a CS introductory sequence, now its own sequence
  - GDD BS has around 20% women, while CS is around 6%
  - First year retention ranges from 85-95% while CS retention is lower at around 65-75%
- Multiple languages learned early:
  - First course Java
  - Second and third courses C#
- Collaboration:
  - Learned early
  - In class exercises
  - Group critiques and analysis
  - Show and tell
- Problems are motivated by real game development issues
- Students claim that the courses are harder than alternatives
Union College

• 5 theme-based intro courses (plus one for engineers), partially supported by NSF CPATH grant, 1 major & 3 minor options

• Language relevant for the theme (Python, C)

• All “studio-style”, combining lecture and hands-on, pair work in class, individual homework

• Assignments based on theme, problems are relevant to students

• Interest in the theme = motivation to overcome “hard” parts

• Outcome – 85 intro students per year from across all disciplines (more with intro for engineers)
• Traditional track to CS 3:
  • Foundational Programming followed by Data Structures
  • Almost entirely white male

• Multimedia track to CS 3 (2 year old curriculum):
  • Interactive Media Computing
  • Socially Responsible Problems
  • 30% female (College is 52% female), ethnic diversity matches College.
  • Scratch in first week to front load programming (like Harvard), followed by Processing for media design emphasis
  • Focus on collaborative support for individual contributions to large project.
  • “Remixing” rather than “lock step” correct solution to single problem.
  • Wikis for public submission of homework, reflective writing, project collaboration.
CS 3 - software engineering and new language (C++)

- Collaboration
  - Semester-long team projects
  - Writing assignments
  - Group assignments and discussions
  - Formal and informal presentations with peer feedback

- Types of problems
  - Challenging
  - Socially relevant
  - Interdisciplinary

- Tools
  - Wiki, Trac, CVS / SVN
Set them up for success – not failure

Questions???

• Choosing the Right Tools

• Peers, Teams, Expertise, Ownership

• What makes a “good” Computer Science problem?

• Don’t Dumb it Down – But They Can!