CS O Materials

CS 0 Courses

- CS Principles
  - Started with support from National Science Foundation (NSF)
- ECS (Exploring CS)
  - Started in Los Angeles (LA) Unified School District
- Other “programming-centric” courses, but not at the AP CS level

A Sample of Books

- “Blown to Bits,” by Abelson, Ledeen, and Lewis
  - Free: www.bitsbook.com

- "D is for Digital," by Kernighan (2011) [Princeton]
- "Fluency with Information Technology: Skills, Concepts, and Capabilities," by Snyder (2010) [U. Washington]
- "Computer Science: An Overview," by Brookshear (2011)
- "Algorithmics: The Spirit of Computing," by Harel with Feldman (2004)

A Sample of Tool-specific Books

- Alice
  - “Learning to program with Alice,” by Dann, Cooper and Pausch, 2011.
- Scratch
  - “Scratch programming for teens,” by Ford, 2008.
- App Inventor
  - “App Inventor: Create Your Own Android Apps,” by Wolber, Abelson, Spertus & Looney, 2011.
    - Free online version: http://www.appinventor.org/projects
- Java
  - “Java: An introduction to problem solving and programming”, by Savitch, 2011.
  - “Introduction to Computing & Programming with Java—A multimedia approach”, Guzdial and Ericson, 2006.
- Python
  - “Python programming for the absolute beginner,” by Dawson, 2010.
  - “Introduction to Computing and Programming in Python—A multimedia approach”, Guzdial and Ericson, 2012.

Other Reading Materials

- Computers are usually in the news, big stories include:
  - Siri (2011) verbally communicates with users
  - Watson (2011) beats human champions on Jeopardy!
  - Google Car (2011) drives itself on urban roads
  - iPhone (2007) allows gestures on touch screen

- Discussing recent advances would be engaging

CS Unplugged

- csunplugged.org
- Activities without computers [more later]

- Computers, like calculators, are tools that take instructions
  - We need to have ideas on how to solve problems
  - Ideas usually don’t need computers

Animation-based Tools

- Building animations/games
- Visually more engaging for learning basic CS concepts
- Examples:
  - Alice [Duke Tutorials]
  - Scratch [Scratch Ed Community]
    - BYOB (SNAP)—Berkeley version with additional features

Phone-based tools

- App Inventor for Android phones
  - Similar to Scratch (with “lego blocks”)

Text-based Tools

- Similar to the “real world”

- Programming languages
  - Java
  - Python

- Programming environment
  - Java: notepad++, blueJ, eclipse
  - Python: notepad++, eclipse

College Pilot: CS 10 at Berkeley

- inst.eecs.berkeley.edu/~cs10
- Lectures: video's are posted
- Textbook: "Blown to Bits" and reading materials are online
- Software: BYOB (a more sophisticated version of Scratch)
- All resources are free!

BJC at Berkeley

- bjc.berkeley.edu
- Beauty and Joy of Computing

- High school adaptation
- Professional development

Pilot Sites for CS Principles

- http://www.csprinciples.org/home/pilot-sites

- Patrick Henry High School (VA)

Exploring CS

- 296-page curriculum
  - http://www.exploringcs.org/wp-content/uploads/2010/08/ExploringComputerScience-v4.0.pdf
  - lesson plans, final projects, scoring rubric

RICS (under development at FIT)

- Repository for Interdisciplinary CS
  - www.cs.fit.edu/~pkc/rics/
- Motivate CS with interdisciplinary problems
  - “Real-world” problems
- small effort Big Effect (seBE)
  - Students work on a small and well-defined problem
  - In the context of a larger problem that has been solved
  - Programs with a small part removed--sample solutions are provided
- Lesson plans with sample discussion points

Free non-credit online university courses

- ocw.mit.edu [2002]
  - MIT Open Courseware—videos of classroom lectures
- see.stanford.edu [2008?]
  - Stanford Engineering Everywhere—videos of classroom lectures
- udacity.com [2012]
  - More personal with built-in interactive exercises
- coursera.org [2012]
  - More personal with built-in interactive exercises
- edx.org [2012]
  - MIT/Harvard/Berkeley edX

- khanacademy.org [2006]
  - High school to college materials

Important Note

- A full CS course would be nice
- If not, incorporating CS materials into existing courses might be appropriate

Questions?