Student Experience in Introductory Computer Science Courses

Jessica Q. Dawson, Meghan Allen
Goals

• understand who takes introductory CS courses, from what disciplines, and why
• evaluate students’ attitudes towards learning CS
• evaluate students’ experience and performance in their introductory CS course.
Why?

1. Inform improvements to existing courses

2. Inform design and measure success of two new introductory courses targeted at non-CS majors
   a) CPSC 103 - Introductory CS Course using Python
   b) Computational Thinking

⇒ Both slated for introduction in 2016/2017
Background

• CPSC 110 – Computation, Programs and Programming
  – **Goal:** introduce students to a systematic method for solving hard design problems.
  – 4 credits; no pre-requisites
  – uses teaching languages (BSL, ISL, ASL), subsets of Racket

• Intended to be widely accessibly
  – required for CS – majors
  – also taken by a wide variety of students in other disciplines
    • But in practice, many nn-CS majors find it to be too intense!

→ In 2015/16 only option for most students interested in CS
Background

• a few other introductory computing courses available

• generally focused on computing for students in specific other disciplines, e.g.,
  – CPSC 301 – Computing in the Life Sciences
  – APSC 160 – Introduction to Computation in Engineering Design
  – EOSC 211 - Computer Methods in Earth, Ocean and Atmospheric Sciences
2015W1 - Outcomes in CPSC 110

• Students from programs outside Faculty of Science
  - often have worse outcomes in CPSC 110
  - withdraw in larger numbers
Methods and measures

• Pre- & post-term surveys (starting in 2015W1)
  – CPSC 110 (& 301)
  – Student attitudes towards CS
  – Computing Attitudes Survey
  – Reasons for taking CS, goals for course
  – Satisfaction with course, perception of developed skills
  – Helpfulness of specific course resources

• Interviews with students who withdraw/fail CPSC 110
  – reasons for taking CS
  – barriers to success in the course
Reasons for tasking CPSC 110

Students are taking CPSC 110 for a wide-variety of reasons besides being required.
Goals students hope to achieve

- Getting a particular grade: 49.5%
- Learning the required skills and concepts: 72.7%
- Staying on top of the course material: 45.1%
- Creating a computer program: 29.5%
- Learning or improving your study skills: 28.3%
- Finding out if you're interested in...: 40.6%
- Working hard: 14.3%
- Making friends: 2.9%
- Other: 2.9%

Many goals focused on general university success than specifically about learning computer science.
Early results: CPSC 110 (2015W1)

What skills do students develop?

• Open-ended survey question in 2015W1

• Top skills students feel they get out of the course:
  – Problem-solving skills
  – Foundational CS concepts (data structures, search, etc.)
  – How to program; how to learn programming languages

• Problem-solving the ‘top’ skill students across disciplines expect to use in their day-to-day lives
  – Many non-CS student do not mention programming, or report being unsure of how they will apply it in the future
For planned CS majors, we see small gains (or no change) towards expert thinking in the 5 different facets of attitudes.
For non-CS majors, we see the opposite trend, with shifts away from expert thinking in most of the different attitudes facets.
Why do students withdraw?

- Small number of interviews conducted so far (n=6)
- Usually withdrawing to reduce their workload, and often for expected reasons, e.g.:
  - difficulty getting use to living away from home, being more independent
  - overwhelmed by amount of work per course (“a lot more than high school”)
  - balancing demanding part-time jobs
Why do students withdraw?

• But why drop CPSC110 and not another course?

• A few early themes that stand out . . .
  
  – Learning goals in CPSC 110 didn’t match the students’ expectations of “what CS is”
    • Unsure what to expect, or expecting to learn more practical and applied skills (e.g., website design, animation, etc.)
  
  – Trouble making friends in the course, finding people to work on problem sets with
    • the “only one of my friends taking CS”
  
  – CPSC 110 usually their highest workload course by far

  – Often leave convinced they “can’t think like a computer scientist”
**EARLY RESULTS: CPSC 110 (2015W1-2)**

**Influence on CPSC 103 design**

- **Goal:** teach students how to take a problem from a discipline of their choice and solve it systematically using computation.
  - 3 credits instead of 4
  - more modest learning goals
  - project that allows a student to connect to a discipline of his or her choice.
  - Uses a language (Python) that non-CS majors might be likely to encounter again in university or at work.
Long-term evaluation plan

• Continue surveys in CPSC 110 & 301 through 2016/17
  – continue interviewing 110 students who withdraw/fail

• Continue to roll findings into course design
  – particularly into CPSC 103 (110 – ‘lite’)

• Conduct surveys in new courses in 2016/17
  – Are the students we are targeting these courses to actually taking them?
  – How do the outcomes of non-CS students compare to CPSC 110? Do they improve for the reasons we hope?
  – Do these new courses improve the introductory experience?