Initial Experiences with a CS + Law Introduction to Computer Science (CS 1)

Robert H. Sloan
Dept. of Computer Science
University of Illinois at Chicago
sloan@uic.edu

Cynthia Taylor
Dept. of Computer Science
University of Illinois at Chicago
cynthiat@uic.edu

Richard Warner
IIT Chicago-Kent College of Law
rwarner@kentlaw.iit.edu

ABSTRACT

We present the curriculum, pilot offering, and initial evaluation of a CS + Law based CS 1 course that was team taught by a Computer Science professor and a law school professor. Relevant legal topics were interwoven through the course. The results from this initial offering suggest that this sort of highly interdisciplinary offering can be successful both in computing education and in making students realize the relevance of Computer Science to the broader world beyond IT.

Keywords

CS + X; Contextualized CS; CS 1; CS + Law

1. INTRODUCTION AND MOTIVATION

Should the FBI be able to force Apple to circumvent the cryptography on an iPhone? Should any government be able to force decryption by any computing company? What rights to use copyrighted digital media do students, professors, and the public have? Did Russia attempt to influence the 2016 US Presidential election by hacking the US Democratic party? Questions at the boundary of Computer Science and Law have never been more pertinent than they are right now.

Within Computer Science education, there has been growing interest in CS + X approaches. Two of the US's largest and most well known CS departments, Stanford and University of Illinois at Urbana Champaign, have introduced a whole series of CS + X majors ranging from CS + Anthropology to CS + Spanish [11, 13].

There have also been several CS + X versions of CS 1 introduced. Guzdial's Media Computation course is at least implicitly a CS + X approach [5,6], and Harvey Mudd's CS 5 Green [4] is explicitly a CS + Biology CS 1. However, there have been very few who have looked at the intersection of *law* and Computer Science (outside of narrow coverage in a Computer Ethics class or a Computer Ethics unit of a senior design course), and as far as we know, none who have looked at CS + Law in the context of an introduction to Computer Science.

We report here on our initial experiences with a CS + Law CS 1 course after its first offering, and give student-generated evidence of its strong and weak points. This course covered as much *computing* material as our other sections of CS 1, but not precisely the *same* material. In particular, there was significant coverage of data analytics, and only light coverage of object orientation and no coverage of recursion. The complete curriculum and materials are available at https://www.cs.uic.edu/CS111Law. The CS + Law section of CS 1 was one of three sections of CS 1 offered in Fall 2016; all three are treated simply as different sections of "CS 1 for CS majors and other interested students."

Some key findings on this course's first offering include:

- On the overlapping Computer Science material, students in the CS + Law section performed roughly as well, and in some cases better, as students in our other two sections (based on common final exam questions).
- The CS + Law section drew a *much* higher percentage students from majors outside of Computer Science or Engineering than our other two sections of CS 1.
- The CS + Law section drew a mildly lower percentage of women than our other two sections of CS 1.
- In a post-course survey, students in the CS + Law section were more likely to respond positively to the statement, "I understand how the material covered in this course relates to society".

2. CS + LAW BACKGROUND

There are longstanding deep connections between law and technology. Patents are explicitly mentioned in the US Constitution, and the first US patent law was enacted by the first Congress in 1790. Today there are several dozen strong journals of law and technology. A bibliography of books written in the past ten years about the intersection of legal and highly technical computing issues in intellectual property, computer security, cryptography, privacy, computer crime, net neutrality, use of data analytics for policing, etc., would surely run to many thousands of titles.

2.1 Why CS + Law is a Good Idea

A law-themed CS course meets two needs. First, the twenty-first century economy needs most, perhaps even all, college educated workers and citizens to know something of college-level CS. Second, addressing critical ethical and public policy questions requires input from people who understand CS and the related ethical, social, and political questions.

ITiCSE '17 July 03-05, 2017, Bologna, Italy

DOI: http://dx.doi.org/10.1145/3059009.3059029

A law-themed course introduces students to both CS and policy issues in a natural and forceful way in the very first CS course.

Almost every CS program says (and generally means) that it wants its students to consider such policy issues very seriously, but it is often difficult to work them into the curriculum any way except with a dedicated course. Not only are some programs reluctant to dedicate a whole course to ethical, social, and public policy issues, but also a separate course isolates those issues from the arts of problem solving and coding. In a law-themed course in contrast, students write and then discuss code that raises public policy questions. This brings the issues alive in a way that can be difficult to do in a separate course.

Bringing the issues alive in this way should make the CS 1 course appealing to a more diverse group of students. Many students enter college with at least some interest in the law as it applies to technology-focused public policy issues, but US law degree is a post-graduate professional degree. At most schools, there are one or two undergraduate courses in Political Science that truly truly cover substantive legal topics. A law-themed CS 1 can draw in majors and prospective majors in Communications, Criminology, Political Science, Management, and Public Health, some pre-law, and some not. It will be doubly useful to them because it provides *two* useful lenses through which to view the world. Just as "[c]omputation is widely accepted as a lens for looking at the world" [3], so too law is a useful lens for looking at public policy questions in general, and public policy questions in electronic security and privacy in particular.

The broader appeal should make CS more welcoming to women. Law school students have been close to gender balanced for decades now. Total US law school enrollment has been at least 40 percent female since the 1985–1986 school year, and has been consistently running between 45 and 49 percent female since 1997–1998 [1]. Fewer women than expected may have enrolled in the Law section because was only added to the course offerings in the Summer, which limited advertising to pre-law students.

3. CURRICULUM

The setting for this curricular experiment is the University of Illinois at Chicago, a large, diverse, public research university. During Fall 2016, we offered three sections of our CS 1 course: The law-focused section, a new biology-focused section inspired by Harvey Mudd's CS 5 Green [4], and a section using media computation that previously had been our only offering of CS 1. Our CS 1 targets Computer Science majors but is open to all; our school also offers other introductory CS courses that are strictly for non-CS majors. For this pilot offering, the law section was capped at 45 students; the new biology section had around 30 students; the media computation section had 160 students. All three sections had the same structure: Two 75 minute lectures a week, and one weekly 50-minute closed lab. All three sections used Python and made some use of peer instruction [2, 7, 9, 10]. The CS + Law section made heavy use of peer instruction.

There is no textbook that covers the precise blend of problem solving, Python programming, and data science that we wanted to cover, much less one with examples coming from law. We used substantial portions of the open source online textbook *How To Think Like A Computer Scientist* [8], and a host of ad hoc materials for other bits and pieces of computing not covered in that text and for all the legal content.

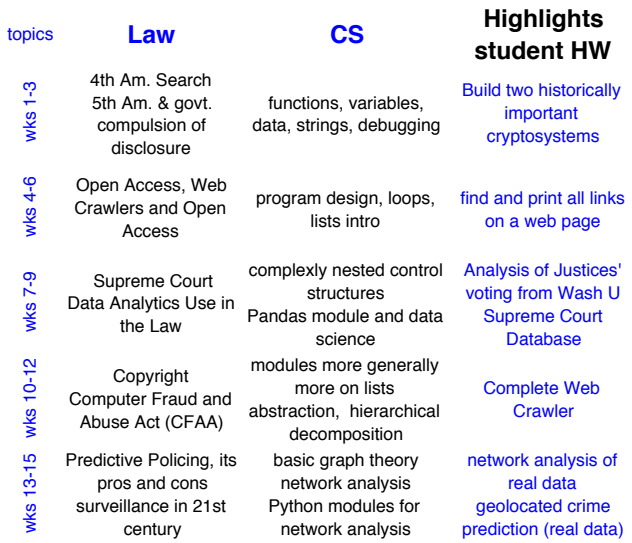

Figure 1: Summary of syllabus, law and computational topics, and a sample of student assignments.

For this pilot offering, we wanted to make sure that the legal material was treated as a first-class citizen. Therefore, the course was team taught by a Computer Science professor and a law school professor, who each have some very basic knowledge of the other's discipline. In the first week or two of the course legal material took up almost as much time as computing material; thereafter legal material took up roughly 20 to 30 percent of class time.

Most weeks the closed lab consisted of a simple three question multiple-choice quiz followed by some simple programming task that perhaps half of the students would finish in the lab. There were eight longer out-of-lab homework assignments; most built on top of one of the labs.

Figure 1 presents an overview of our syllabus, broken up into five modules, together with a small sample of the lab and homework assignments.

3.1 Combining Law with CS

Our primary goal was to teach a CS 1 oriented around creative problem solving, and a sampling of legal topics with connections to computing. There were no legal topics that we "had" to cover, except that, as we discuss in Section 3.2, we wanted to include some data science partially because of the law theme and partially because we believe it is a very important contemporary computing topic.

Three assignments combining law and CS were particularly successful. First, early in the course, after an introduction to strings, students had to write a Caesar and a Vigenère cipher. The related law discussion concerned Fourth Amendment protections against government searches and Fifth Amendment protections against government demands for encryption keys. Both discussions can easily enter legal thickets of interest primarily to legal experts, but the assignment tied the discussion to students' work, and that gave the discussion a very useful concreteness. In this way, the assignment illustrated early on how the law and coding themes go together.

Those two cryptosystems are lovely assignments for sev-

eral weeks into any Python-based based CS 1 that chooses to start with strings. The Vigenère cipher is built on top of the Caesar cipher in a moderately complex way that makes an excellent example of building more complicated functions out of simpler ones. In general, we believe that assignments that produce actual working software that can do something real (though not necessarily at scale or in a 21st century way), such as matching encrypt and decrypt functions, are particularly appealing to students.

The second assignment was to write a web crawler that harvested email addresses. The task is a complex one for a CS 1 targeting both majors and non-majors. We covered a bit of the material right after the cryptosystems, and then spread the remainder over a few weeks in the second half of the course. This provided a unifying theme and gave ample time to discuss the related legal issues of unauthorized access under the Computer Fraud and Abuse Act (CFAA) and copyright violations. We were discussing these issues while students wrote code that could potentially be used to commit both CFAA violations and copyright violations. (We cautioned students to use that crawler *only* on the test website we specially constructed for them). The class discussion around the issues owas particularly robust.

The third assignment concerned data analysis using the Supreme Court database (http://scdb.wustl.edu/). We introduced the Pandas module (http://pandas.pydata.org/), which provides R-style ability to analyze very large dataframes in Python, for this purpose. Data analysis is rapidly growing in importance and acceptance in the legal world, so the assignment showed students they had acquired skills that they could put to real world use, and it underscored the connection of technological choices and public policy issues.

The mix of law and computer science underscored and illustrated the importance of each topic and made the teaching experience particularly satisfying.

3.2 (Other) Differences from a Classic CS 1

We made a deliberate decision to introduce some elements of beginning data science into our course. Data science is a field of growing importance, with strong roots in computer science, so it is, in our opinion, a natural fit in any CS 1 course. Additionally, data science is of interest to a growing minority of practicing lawyers—it seemed right to introduce our students to a tool that the most tech-oriented lawyers will use. Finally, data science is really impressive to beginning students, in that they are able to read in a data set with a million rows and dozens of columns and process it on their laptops. Seeing a data set that Excel can open only after 5 minutes, if at all, be read in by a Python program using the Pandas module in 5 seconds is a very forceful lesson about the power of being able to code, instead of always having to rely on other people's code. We also believe it is really powerful to work with real data in a first course, instead of "toy" data.

We had a total of three assignments that were heavily or exclusively data science. One was the analysis of the Supreme Court database. A second was a module on predictive policing. Students used a very large, real-world database of urban crime and made their own predictions based on it. A third was a classic (very simple) network analysis.

There were trade-offs in what we covered. The network analysis meant we had to introduce basic ideas of graph theory. Of course presenting legal material took up some of the course's time that would be spent on problem solving and coding in a traditional CS 1 course. We did not cover recursion at all, and we did not cover the design of classes at all, and touched on object orientation only lightly in terms of using methods from built-in classes such as strings. However, research by Tew et al. suggest that even when exposed to different material in CS 1, students' knowledge of introductory topics is likely to converge after completing the same CS 2 course [12].

Additionally, our evaluation, discussed in the next section, showed that students found our section to be considerably more work than the other sections of CS 1. As a result, we may have to reduce the total amount of material we give to students in future offerings of this course.

4. EVALUATION

As mentioned, during Fall 2016, we offered three sections of our CS 1 course: The law-focused section, a new biology-focused section, and a section using media computation that previously had been our only offering of CS 1.

We surveyed students in all three courses before and after the course, and had five identical common short questions across all three final exams, and one similar in spirit, different in details, longer coding question across all three final exams. We also had access to the regular end-of-term course evaluations for our law-focused section, and demographic information for all three courses.

4.1 Student Learning

We asked five identical multiple-choice questions on all three sections' final exams: three on control structures, and two on function calls and parameters. The students in the CS + Law section did significantly better over all, averaging 77 percent correct responses to the five questions, versus 61 and 62 percent correct in the other two sections.

We also asked one broadly similar coding question on all the sections' final exams. It asked students to produce a weighted sum of two sequences, with some sort of limiting rule of the form, "If the weighted sum is greater than x in absolute value, use x with the appropriate sign as the value." For the law section, this was posed as two hash tables with the same set of keys, and the values being different indicators for predictive policing. For this question, students in the law section performed the worst, earning on average 61 percent of the points, versus 71 percent in both of the other sections.

A tentative conclusion would be that the CS + Law section was at least as effective as the other sections in teaching Computer Science concepts, but because the CS + Law section had students spending a fair amount of time on data science explorations, time taken from programming, that students were somewhat weaker at writing code.

4.2 Student Comments

Students generally found the instructional approach worthwhile. Here is a selection of comments in response to "Please comment on specific characteristics of the course that were most beneficial to you" in our institution's regular post-course course evaluation:

- "Understanding of Python in an exciting way."
- "It was a good use of Python, and it was interesting in building a web crawler."
- "It was interesting to see real world data implemented with the course assignments."

Table 1: Ethnicity, gender, and CS major data in percentages, compared to two other Fall 2016 sections of CS 1, and the Fall 2015 single CS 1 (media computation) section.

	Law	Media	Bio	Fall 15
Asian	31	30	29	39
Black	4	6	11	6
Hispanic	20	22	30	15
International	2	5	11	6
Multiracial	7	2	4	2
White (non-Hispanic)	33	34	15	30
Female	18	22	22	25
Male	82	78	78	75
CS Majors	53	87	74	NA

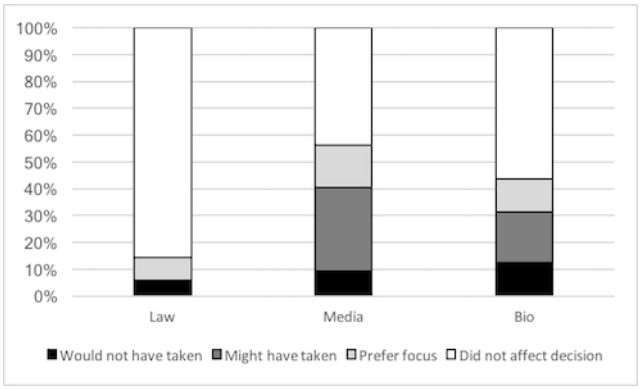

Figure 2: Answers to the question "How important was the Law/Media Computation/Biology focus in you deciding to take this course? A. Would not have taken an introductory computer science course with a different focus B. Might have taken a different introductory computer science course, but focus was the deciding factor C. Would have taken a different introductory computer science course, but prefer one with this focus D. Did not factor into decision to take an introductory computer science course."

- "Very lively, had very interesting ideas, and was very interactive with the class."
- "Working with real data and making a web crawler."

4.3 Course Demographics

We obtained complete demographic information for the students from our institution. As shown in Table 1 the ethnicity distribution of the students enrolled in the Law section was very similar to that of Media Computation section. The Law section stands out for having *many* more non-majors, and mildly fewer women.

4.4 Pre-course Survey

Before the course started, we surveyed the students to see why they were taking the course, and what their initial perceptions of computer science were.

Reasons For Taking Course.

We surveyed students on how important the particular focus of the course (law, biology or media computation) was to their decision to take a CS 1 course. As shown in Figure 2, the Law focus was significantly less important to students than the Media Computation or Biology focus. This was a statistically significant difference from the Media Computation course (p-value = .0008). (All p-values were calculated using a two-sample t-test.) This may be partly because the Law section was added as a new course over the summer, and we were unable to advertise it to pre-law and other students as much as we would have liked.

The Law section of CS 1 had both fewer students who had already declared a CS major (see Table 1) and significantly fewer students who entered planning to major in Computer Science than in the traditional Media Computation section (see Figure 3). Using a 5 point Likert scale, with "Strongly Agree" coded as a 1, and "Strongly Disagree" coded as a 5, Law had a mean of 1.42 versus Media Computation's mean of 1.06 (p-value = .046). (A similar difference is also observed in our other targeted section, Biology, which had a mean of 1.5.) This difference may be due to the Media Com-

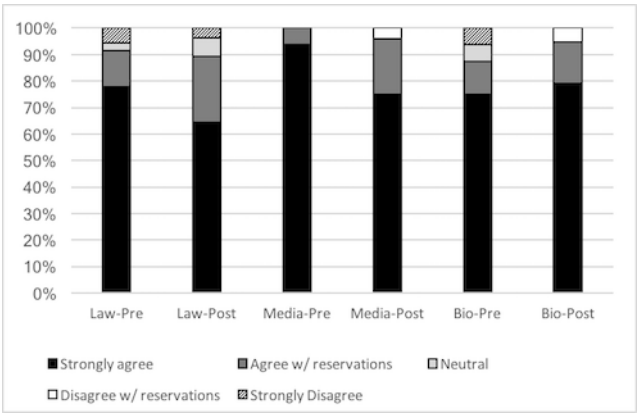

Figure 3: Responses to the statement "I plan to major in Computer Science", before and after taking the course. "-Pre" indicates responses from the pre-course survey, and "-Post" indicates responses from the post-course survey.

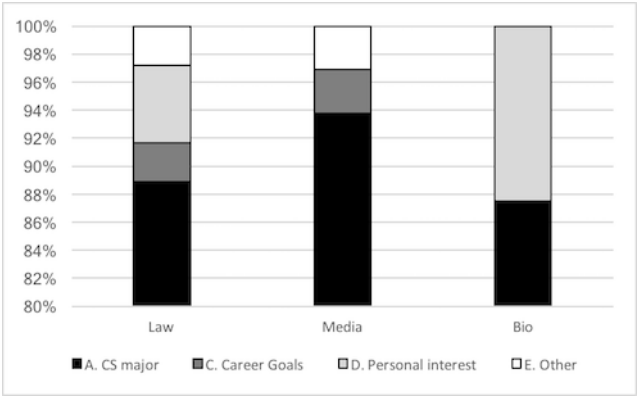

Figure 4: Responses to the question "Reason for taking course: A. Interest in CS major, B. Relevant to non-CS major, C. Relevant to career goals unrelated to major, D. Personal interest, E. Other."

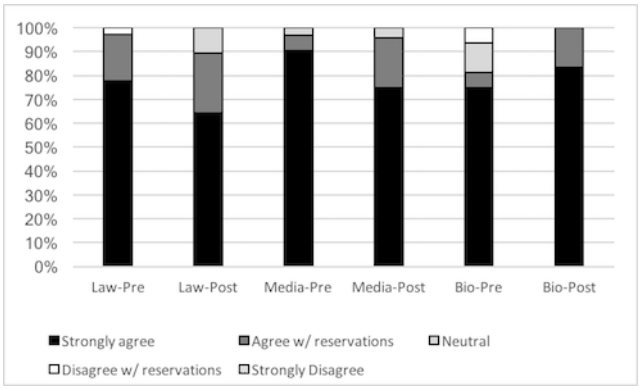

Figure 5: Responses to the statement "Knowing programming will help me earn a living."

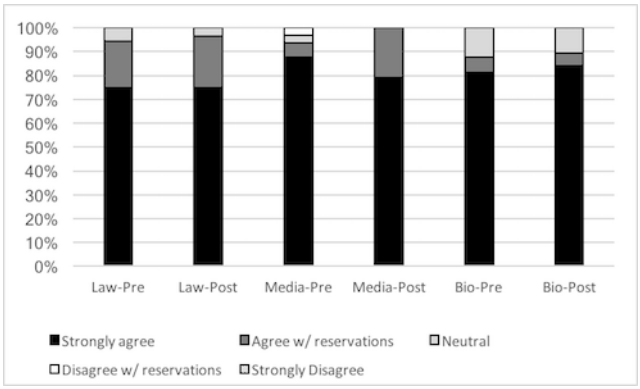

Figure 6: Responses to the statement "Computer science is a worthwhile and necessary subject."

putation section being the traditional section for our majors to take, or our new targeted sections appealing to students who were less certain they wanted to major in CS. As shown in Figure 4, both targeted sections had more students taking the course for reasons other than planning to major in CS.

Student Opinions on Computer Science, Pre-Course.

Students in the Law section were consistently less likely to agree with statements like "Knowing programming will help me earn a living" (Figure 5, mean of 1.28 vs 1.13), "I'll need programming for my future work" (mean of 1.306 vs 1.25), "Computer science is a worthwhile and necessary subject" (Figure 6, mean of 1.31 vs 1.22), and "I will use programming in many ways throughout my life" (mean of 1.44 vs 1.31), although not to a statistically significant extent. This may indicate students in the Law section came into the course feeling less positive about Computer Science in general than students in the Media Computation section. Students in the Biology section were also less likely to agree with these statements, indicating that our alternative focused sections attracted a population of students who were less sure about Computer Science.

4.5 Post-Course Survey

After the course completed, we surveyed all three sections on their perceptions of computer science and feelings about the course.

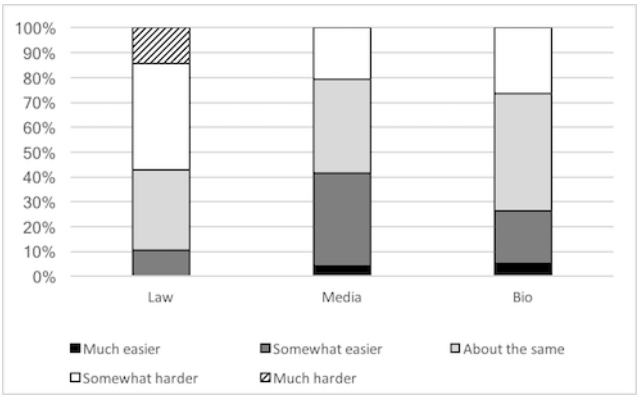

Figure 7: Student responses to the question "Compared to other classes, how difficult do you find this class?"

Likelihood to Major, Post Course.

While before the course students in the Law section were significantly less likely to agree with the statement "I plan to major in Computer Science" than students in the Media Computation section, after the course there was not a statistically significant difference between responses from the students in the Law section, and responses from the students in the other sections (shown in Figure 3, Law mean of 1.54, Media computation mean of 1.33, Biology mean of 1.31). In both the Media Computation and the Law section students had more reservations about majoring after taking the course than before taking it, possibly due to many of them being exposed to what Computer Science actually involves for the first time.

Student Reactions to Assignments.

The Law section found their assignments significantly more difficult than the Media Computation or Biology sections, as show in Figure 7. Using a 5 point Likert scale to answer the question "Compared to other classes, how difficult do you find this class?", with "Much harder" coded as a 5, and "Much easier" coded as a 1, the Law section had a mean of 3.61, the Media Computation section had a mean of 2.75 (p-value = .0008), and the Biology section had a mean of 2.95 (p-value= .014). This may be related to the high proportion of Law students entering the course with non-engineering majors, or students entering the course less sure of their intention to be Computer Science majors. However, as this is the first offering of this course, assignments may have genuinely been more challenging or time consuming than the equivalent assignments in the Media Computation or Biology sections.

Students in the Law section agreed with the statement "I understand how the material covered in this course relates to society" more than students in either the Media Computation or the Law section. On a 5-point Likert scale with "Strongly agree" coded as a 1, students in the law section had a mean of 1.5, versus 2.09 in Media Computation (p-value = .051), and 2.11 in Biology (p-value = .16) (shown in Figure 8). This seems likely due to the fact that one of the main focuses of the Law section was the impact of Computer Science on society and the law, and many of the assignments looked at socially relevant problems.

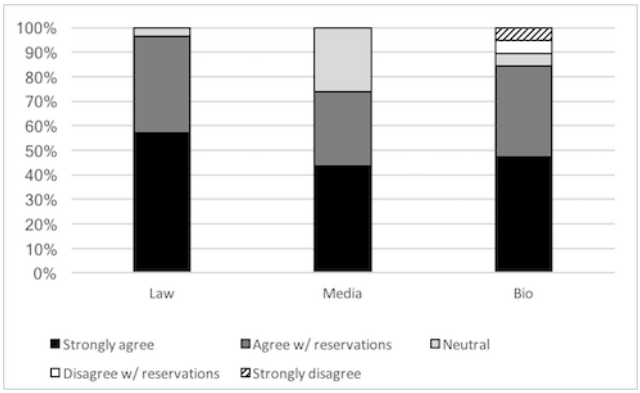

Figure 8: Student responses to the statement "I understand how the material covered in this course relates to society."

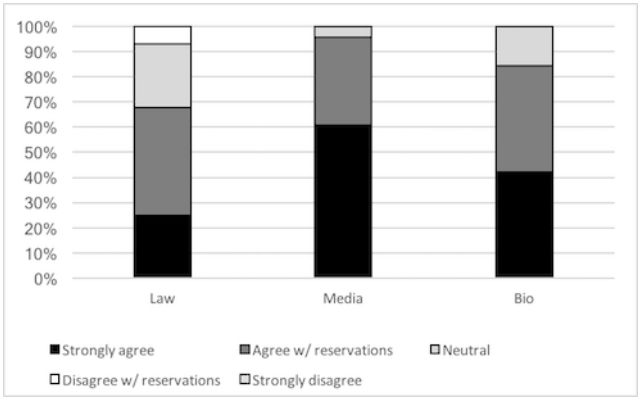

Figure 9: Student responses to the statement "Assignments increased my interest in the subject matter of computing."

In both of our specialized sections, students agreed less with the statement "Assignments increased my interest in the subject matter of computing". The Law section had a mean of 2.54, while Biology had a mean of 1.90 (p-value = .089) and Media Computation had a mean of 1.48 (p-value = .002). It may be that the application of Computer Science to a specific topic resulted in students feeling less focused on CS as its own subject. Alternatively, it may be that the students in the Media Computation class, who were overwhelmingly planning on a CS major when they entered, were simply more positive on any measure of interest in computing.

5. CONCLUDING THOUGHTS

A law-themed introduction to computer science can be successful both in computing education, and in making students realize the relevance of Computer Science to current social and public policy issues. Our initial data suggests that students find seeing the relevance interesting and that it motivates some of them to take more computer science courses. We plan to more heavily advertise future offerings to pre-law students, and hope as a result to have a greater percentage of women in the course.

6. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. DUE-1612455.

7. REFERENCES

[1] American Bar Association. First year and total J.D. enrollment by gender 1947 - 2011. Online at http://www.americanbar.org/content/dam/aba/administrative/legal_education_and_admissions_to_the_bar/statistics/jd_enrollment_1yr_total_gender.authcheckdam.pdf.

[2] C. H. Crouch and E. Mazur. Peer Instruction: Ten years of experience and results. *American Journal of Physics*, 69(9):970–77, 2001.

[3] P. J. Denning. The profession of IT: Beyond computational thinking. *Communications of the ACM*, 52(6):28–30, 2009.

[4] Z. Dodds, R. Libeskind-Hadas, and E. Bush. When CS 1 is Biology 1: crossdisciplinary collaboration as CS context. In *Proceedings of the 15th Annual Conference on Innovation and Technology in Computer Science Education*, ITiCSE '10, pages 219–223, 2010.

[5] A. Forte and M. Guzdial. Computers for communication, not calculation: Media as a motivation and context for learning. In *Proc. 37th Hawaiian International Conference of Systems Sciences*, 2004.

[6] M. Guzdial. A media computation course for non-majors. In *Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education*, ITiCSE '03, pages 104–108, 2003.

[7] C. B. Lee and B. Simon. Peer Instruction for computer science website. Online at http://www.peerinstruction4cs.org/.

[8] B. Miller, D. Ranum, J. Elkner, P. Wentworth, A. B. Downey, C. Meyers, and D. Mitchell. *How to Think Like a Computer Scientist: Interactive Edition.* Runestone, 3rd edition, 2014. Online at http://interactivepython.org/runestone/static/thinkcspy/index.html.

[9] B. Simon, M. Kohanfars, J. Lee, K. Tamayo, and Q. Cutts. Experience report: Peer Instruction in introductory computing. In *Proceedings of SIGCSE 2010*, pages 341–45, 2010.

[10] J. Spacco, J. Parris, and B. Simon. How we teach impacts student learning: Peer Instruction vs. lecture in CS0. In *Proceedings of SIGCSE 2013*, pages 41–46, 2013.

[11] Stanford University. Joint majors: CS + X. Online at https://majors.stanford.edu/more-ways-explore/joint-majors-csx.

[12] A. E. Tew, W. M. McCracken, and M. Guzdial. Impact of alternative introductory courses on programming concept understanding. In *Proceedings of the First International Workshop on Computing Education Research*, ICER '05, pages 25–35, 2005.

[13] University of Illinois at Urbana Champaign. CS + X degree programs. Online at http://www.cs.uiuc.edu/academics/undergraduate/degree-program-options/cs-x-degree-programs.