A Biology-themed Introductory CS Course at a Large, Diverse Public University

Tanya Berger-Wolf, Boris Igic, Cynthia Taylor, Robert Sloan, Rachel Poretsky
University of Illinois at Chicago
Chicago, IL
{tanyabw,boris,cynthiat,sloan,microbe}@uic.edu

ABSTRACT

We present the curriculum and evaluation of a pilot Biology-themed CS1 course offering at a large public university. Inspired by Harvey Mudd's CS 5 Green, we adapt CS1 + Bio to fit the needs of our student body, which is much more typical for those US institutions that produce the bulk of the nation's CS undergraduate degrees. This course was team-taught by a computer science professor and a biology professor, and combined typical CS1 topics with relevant biology content. Our initial offering attracted students who would not otherwise have taken CS1, and was the only one of our three CS1 courses where more students reported planning to major in CS after the course than before it.

CCS CONCEPTS

• **Social and professional topics** → *Computer science education*;

KEYWORDS

CS1, CS + X, Contextualized CS, CS1 + Biology

ACM Reference format:
Tanya Berger-Wolf, Boris Igic, Cynthia Taylor, Robert Sloan, Rachel Poretsky. 2018. A Biology-themed Introductory CS Course at a Large, Diverse Public University. In *Proceedings of SIGCSE '18: The 49th ACM Technical Symposium on Computer Science Education, Baltimore , MD, USA, February 21–24, 2018 (SIGCSE '18),* 6 pages.
https://doi.org/10.1145/3159450.3159538

1 INTRODUCTION

This work looks at an introductory Computer Science course with a biology theme developed by adapting Harvey Mudd College's "CS 5 Green" introductory course [1, 8, 9] for use at the University of Illinois at Chicago (UIC), a large, public, diverse research university. Our goals include enabling biology majors and minors to use the power of computing and computational thinking, and attracting more women into computer science.

Alvarado et al. [1] showed CS 5 Green to be highly effective in drawing women into Computer Science *in the context of Harvey Mudd*, a small, private, highly selective, and STEM-focused college. According to its Common Data Set [10], it has a total of 804 students across all years and fields, and three-quarters of its entering students have either an ACT score of at least 33, or an SAT Math score of at least 740. According to *US News and World Report*, Harvey Mudd's acceptance rate was 13 percent in 2015 [17]. Anecdotally, Harvey Mudd has very few biology majors.

Elite STEM schools are outstanding places to start STEM education experiments. However, if we really want to change biology education, we must go to schools that have large numbers of biology majors. UIC has between 1500 and 2000 biology majors, and just over 1000 CS majors, each number being more than the entire enrollment of Harvey Mudd College. To change CS education at the national level, and especially to make major progress on the dismal nationwide statistics on the percentage of women in CS, we must eventually make changes at the schools that produce the most bachelor's degree graduates. The most recent Taulbee Report tells us that public, Ph.D.-granting universities now average nearly 900 Computer Science majors per department, split between about 725 "majors" and 150 "pre-majors" [23]. That is, the average public Ph.D.-granting institution has more CS majors than Harvey Mudd has total students.

CS 5 Green requires substantial reworking to be really useful for such schools. In its original form, the course is aimed at very strong students who are ready, starting with absolutely no CS background, to cover in a single semester, among other things: recursion, recursion on trees, memoization, and computability theory. For many students at our broad set of target schools, this topic list is somewhat ambitious, given that we must simultaneously teach Python programming, as well as a fair amount of biology to a diverse student population with disparate levels of experience.

The Harvey Mudd College context is different from most other schools in two more ways, as well. First, Harvey Mudd is a school targeting the STEM disciplines, and the overwhelming majority of all Mudd students major in a STEM field. Second, every student at Harvey Mudd is required to take an introductory CS course, regardless of major. Most institutions do not require such a course of every undergraduate—ours certainly does not. While today we and many, many other CS programs are feeling overwhelmed by the overall demand for our courses, we nevertheless want to have an introductory CS course, or an array of introductory CS courses, that appeals to a broad range of students. While we may have difficulties meeting the demand for courses, it is crystal clear that it is an advantage to our institution's students if *all* of them take at least one CS course. We expect that a CS+Bio intro course aimed only modestly more broadly than Harvey Mudd's course ought to attract students who would otherwise never take such a course at

ACM ISBN 978-1-4503-5103-4/18/02…$15.00
https://doi.org/10.1145/3159450.3159538

other institutions with a wide mix of students and no campus-wide CS requirement.

2 WHY BIOLOGY?

According to *Vision and Change in Undergraduate Biology Education: A Call to Action* [2], the study of modern biology requires more emphasis on modeling, computation, and data analysis tools than ever before. Undergraduate biology students are expected to acquire knowledge of related disciplines, specifically including computer science. Biology has long been on a path to becoming quantitatively and computationally intensive, yet undergraduate biology curricula have not kept up with this new reality [3, 14].

Many graduates with degrees in biology lack the training in analytical and information-processing skills needed to tackle growing and increasingly complex datasets. These skills often constitute basic requirements for jobs in industry or academic careers. As a result, the inclusion of computer programming in biology coursework is a developing national trend for undergraduate education, endorsed by the NSF and AAAS [2], and used in biology programs nationwide.

There is considerable demand among some of our Biology department undergraduates for greater exposure to computer science and, unlike the situation at Harvey Mudd College, many of our entering students are unaware of the relevance of computational biology and bioinformatics. Providing our Biology department undergraduates with basic programming skills will make them more competitive and prepared for the workforce.

3 COURSE

In this section, we describe the course content and instructional methods.

3.1 Course Content

To ensure the fit of this course for our students, we built on the syllabi of the existing introductory courses at our institution in Biology and Computer Science, ensuring that no prior knowledge in either subject is required. As a starting point, we also used material from the Harvey Mudd CS 5 Green, and the Rosalind [20] platform for learning bioinformatics.

The combined syllabi of our two-course introductory Biology sequence include: processes of cellular and organismic function, cell structure, respiration photosynthesis, molecular genetics and development, structure, and physiology of plants & animals in Bio 1; and in Bio 2: species richness, population biology, community ecology, biogeography, biogeochemical cycles, conservation biology, speciation, and extinction.

Our CS+Bio course includes more biology background and concepts than the Harvey Mudd CS 5 Green course. We begin the semester by defining life and discussing the structure and function of DNA, RNA, and proteins. The reasons for this include the fact that many of our CS students enter the course with little to no knowledge about biology, as well as our hope that this course will at least partially fulfill some of the learning goals of our introductory biology sequence, or reinforce them. To aid this goal, we provide lectures and readings in biology, in addition to the computer science text.

Our standard CS1 course syllabus includes topics in encoding of data, variables, data types, basic statements and operators, simple input and output, while loops and relational operators, method calls and parameters, `for` loops, `if` statements, Boolean operators, return values from methods, and usage of basic data structures (e.g., lists).

Assignments and labs closely intertwined this biology curriculum with the standard CS1 topics (Table 1). Biologically meaningful datasets, including real and simulated nucleotide gene sequences, genotypes, and genomes, provided the source data for a variety of exercises packaged into weekly labs and three larger projects, following the CS curriculum in a manner otherwise nearly identical to our standard CS1. As an instructive example, we spent one biology lecture on the theory of molecular clock and a historical overview, and then used the same hemoglobin sequences (α-globin & β-globin) as Zuckerandl, Jones, and Pauling [22] for CS material on file I/O and a variety of basic string manipulations, as well as molecular distance calculations.

Our CS+Bio syllabus is shown in Table 1. Details on the current or most recent offering of the course are available at: https://www.cs.uic.edu/CS111Green/, and on the Fall 2016 offering, described in this paper at: https://www.cs.uic.edu/CS111GreenFA16/. Interested readers are encouraged to contact the authors for access to full course assignments and more information. The course syllabus followed our standard CS1 course very closely, with the selection from the standard biology syllabus that best matched the CS material, while creating a coherent and self-contained educational unit.

3.2 Adaptation and Differences from CS 5 Green

As mentioned in the introduction, the main goals of our course, and the motivation behind modifying the course design of Harvey Mudd's CS 5 Green, were to serve a student population which is typical for a public Research 1 university, without extremely selective admission, with many students entering with very little or no CS background, and no requirement of a CS intro course for the biology major. Thus, our course aimed to closely follow the existing intro curricula in CS and Bio, which are tuned for our students and would appeal as an elective alternative for biologists (and other majors).

Harvey Mudd's CS 5 Green is listed as covering: "data, lists, and functions", "recursion and functional programming", "combination circuits", "memory, registers and machine language", "the imperative paradigm: loops and accumulators", "maintaining modularity!", "algorithm design: the use-it-or-lose-it idiom", "2d data, memoization and algorithm efficiency", and "algorithm complexity and uncomputability" [8]. Of these topics, the CS1 course at our institution covers "data, lists and functions", "the imperative paradigm: loops and accumulators", and perhaps, if we are being generous, some algorithm design. As a result, the Harvey Mudd assignments and lesson plans were well out of scope for our CS1 course, after the initial introductory material.

We used only Parts 0 and I of the textbook created for the Harvey Mudd CS 5 Green course [12], which is about a third of the material covered by that book. This represented about half of the material

Biology	CS	Example Assignments
Intro to Bio	Intro to CS	
DNA	Variables and statements, Python syntax	(1) Write a program that sets the number of bases of each type (A, C, T, G) into a string (DNA molecule), and then computes and prints its length. You should have variables for the number of each nucleotide, and your program should print the total number. (2) Create a string variable of any length that contains the letters A, C, T, and G. Print a string that is a tandem duplicate of the original string.
Genes, Cells, Central Dogma of Molecular Biology	Sequences, lists, indices, and slicing	Find the first three base pairs of a given gene. Use the codon translation table to find the amino acid for which it codes. Print the first nucleotide in *each* codon of the gene, as a single string.
GenBank	File I/O and manipulation	Access *GenBank* [5]. Find the nucleotide sequences for the human α-globin (HBA1) and β-globin (HBB). Create a file that contains these two DNA sequences, each in a separate line. Add lines that contain each of these two gene sequences, but in reverse.
GC percentage of DNA, transcription, strand orientation, ORF	Functions	Write a simple gene finder with a function that calls other functions, and then use it to find reading frames in a *Salmonella* sequence (obtained from GenBank).
Pathogenicity	Control flow, `if/else`, loops	Write a function that takes four file names (provided), reads the sequence contained in each, then compares sequences at each position, and reports the overall pairwise percentage differences.
Heredity, population genetics	Random numbers	Write a simple population simulation which illustrates Mendelian genetic inheritance and its properties. Create a founder population, randomly mate them with other individuals to produce offspring with a genetic makeup that follows Mendel's laws, thus creating the next generation. Do this for many (N) generations, and calculate the frequencies of different genotypes in the population, following how those change with every generation.
Models of sequence evolution, phylogenetic trees	Dictionaries and arrays, plotting	Write a simple simulation of DNA genome evolution for a group of virtual species. Begin by simulating the first virtual organismal genome of arbitrary length with arbitrary probabilities. Next, implement an algorithm that allows your species to give birth, become extinct, or neither, and undergo mutation in each time step. Finally, implement a procedure to calculate differences between your evolved species' genomes.

Table 1: The CS+Bio Syllabus, listing the topic correspondence between the two disciplines, as well as example assignments.

covered in our class. However, we spent significantly more time covering biological material in class, and covered the computer science content more slowly. In addition, basic operations of File I/O are not explicitly covered by Harvey Mudd course, and the subject of random numbers, necessary for population biology and evolution (or any biology, really) is covered only minimally. Overall, even for the parts of the course which overlapped with the Harvey Mudd course material, we significantly modified the lecture notes and examples to reflect the biological content.

We matched the coverage of introductory biology, but generally provided more detail, and diverged from the Mudd CS 5 Green to follow our standard biology curriculum more closely. Specifically, we progressed along the biological hierarchy, from sequences to genes, to populations, and species (and groups of species). The major changes in our course included coverage of genome structure, molecular evolution (molecular clocks, DNA sequence models, etc.), and expanded population genetics.

3.3 Team Teaching and Peer Instruction

Our course was team-taught by two instructors, one each from the Biology and Computer Science departments. After initial experiments with the timing and combining the biology and computer science lecture material, and after receiving feedback from the students, we settled on lectures alternately devoted to either computer science or biology material. Students explicitly preferred this and noted that switching in the middle of the lecture from one field to another was jarring and disruptive. For practical reasons, we continued to use the split lecture arrangement for the first week of class, for the purpose of introduction and initial material exposure.

Both instructors were present for all lectures (unless precluded by travel or illness).

Peer-instruction-based materials were developed for the course. Peer instruction is a pedagogical technique, which has been emphasized in physics for over 20 years, and is well documented as increasing the students' conceptual understanding of topics [6]. It is increasingly used in computer science [7, 13, 15, 16, 18, 21]. Under peer instruction, throughout the lecture, students are given a multiple-choice question, with one correct answer and two to four distractor answers. They first vote individually on an answer, typically by using an electronic clicker device [4]. They then discuss the question in small, assigned groups, and after a brief discussion, the group votes on a single answer. Peer instruction is effective in developing cross-disciplinary communication skills in an interdisciplinary setting [11], one of the goals of the course.

3.4 Student Recruitment

Students were recruited from three sources: (1) incoming engineering (including CS) majors could take this course instead of our regular CS1, (2) incoming biology majors were able to take the course as an elective, and (3) all Honors College students were able to take the course as an honors elective. The course was open to every student at our institution as a free elective. Undergraduate advisors in three colleges (Engineering, Liberal Arts & Sciences, Honors) received information about the course.

4 EVALUATION

During Fall 2016, we offered three sections of our CS 1 course: The biology-focused section, a new law-focused section [19], and a section using media computation that previously had been our only offering of CS1. The biology section of CS1 had 27 students in this initial offering, in comparison with 45 students in our law section, and 158 students in our media computation section.

All sections were offered in the standard CS1 format for our institution: two lecture sections of an hour and fifteen minutes each a week lead by the course instructors, as well as an additional weekly hour long lab section lead by a TA. All instructors and TAs had office hours available to the students, and additional general CS tutoring hours were available for all the students.

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 longer coding question across all three final exams. We also had access to the regular end-of-term course evaluations for our biology-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 + Bio section performed comparable to the other two sections, with 61%±15 average overall answering questions correctly, compared to 62%±23 for the general section and 77%±6 for the law section.

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 biology section, this was posed as following: Write a function combineProb, which takes two dictionaries of DNA probabilities as inputs and returns one dictionary of DNA probabilities which is the average of the input dictionaries. That is, for all the dictionaries the keys are the nucleotide letters and the values are the probabilities of that nucleotide. For this question, students in the biology section (earning on average 71%) performed similar to the general section (71%) but better than the law (61%).

A tentative conclusion would be that the CS + Bio section was at least as effective as the other sections in teaching Computer Science concepts.

Table 2: Ethnicity, gender, and CS major data in percentages, compared to two other Fall 2016 sections of CS1, and the Fall 2015 single CS1 (media computation) section.

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

4.2 Student Comments

Students generally found the instructional approach worthwhile and particularly enjoyed the active learning approach. 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:

- "Ideal learning environment and ideal instruction for introduction to Python/programming."
- "Excellent overall instruction, much greater than anticipated. Instructor catalyzed new found interest in bioinformatics. Optimal integration of biological and computer sciences."
- "Detailed slides with perfect amount of information."
- "Being able to discuss solutions with my table and then learn if it is correct."
- "We would go through some programs as groups and regroup as an entire class."
- "The pace that [instructor] taught the biology portion of the class was at a good pace."

4.3 Course Demographics

We obtained complete demographic information for the students from our institution. As shown in Table 2, the biology section had more black and Hispanic students, and fewer white students. While the population differences between the courses were not statistically

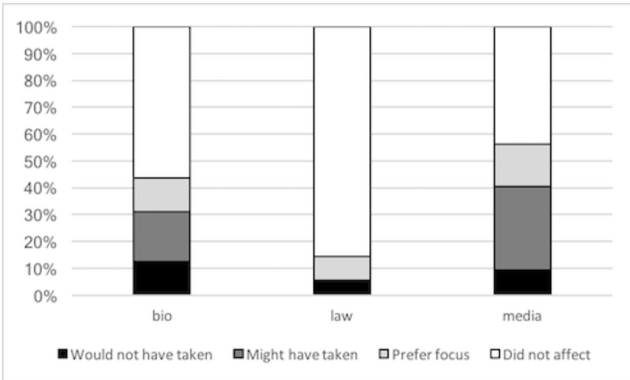

Figure 1: Answers to the question "How important was the Law/Media Computation/Biology focus in you deciding to take this course? A. Would not have taken an introductory computer science course with a different focus B. Might have taken a different introductory computer science course, but focus was 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."

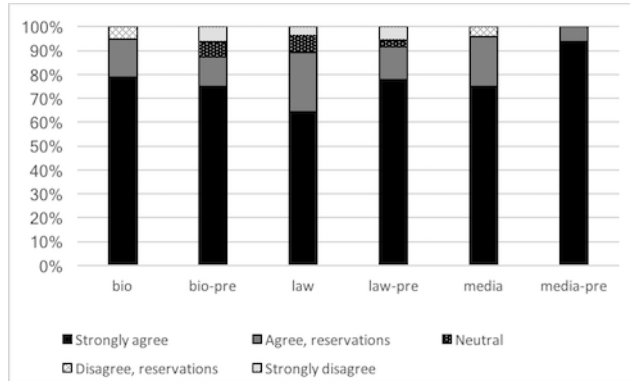

Figure 2: Responses to the statement "I plan to major in Computer Science", before and after taking the course. "-Pre" indicates responses from the pre-course survey.

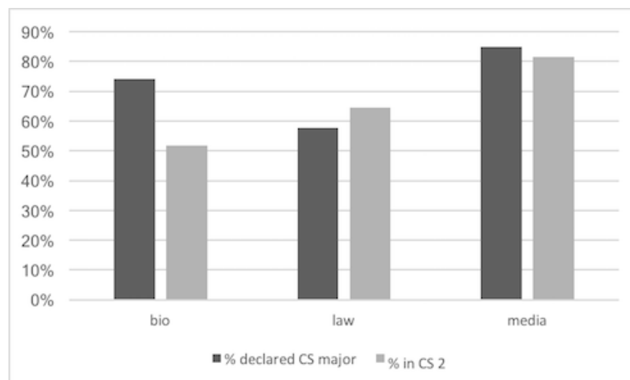

Figure 3: Percentage of each class to major, and to take CS 2.

significant, this is a promising sign towards using biology as a target domain to attract diverse students.

Unfortunately, this section did not succeed in recruiting any more women than any of our other CS1 offerings. This may be partially due to the fact that the course was developed relatively late, and we were not able to advertise it to Biology students before the end of Spring semester. We hope future offerings with more targeted recruiting will be able to attract more women. However, it is possible that without being a requirement, this course will not be able to recruit more women to the CS major.

4.4 Survey Data

We now examine results from a pre- and post-course survey given to students in all three of the CS1 sections. This survey consisted of a series of five-point Likert scale questions aimed at assessing student's feelings about the importance of computer science, and their computer science related self-efficacy. Survey data was evaluated using the Kruskal-Wallis test to determine if there was a statistically significant difference between the three groups, followed by pair-wise Mann-Whitney tests, if a difference was found.

4.4.1 Importance of Biology Focus. The biology focus was relatively important to students in choosing to take this course, as shown in Figure 1. Fifty-six percent of CS 1 + Bio students said the biology focus did not factor into their decision to take the course, compared to 85% of law students and 44% of media computation students. The biology section also had the highest percentage of students who would not have taken a CS course without the focus at 13%, compared to 6% in the law section and 9% in the media section. There was a statistically significant difference between biology and law courses (p-value=0.00038), although not between biology and media computation.

4.4.2 Likelihood to Major. Student agreement with the statement "I plan to major in Computer Science" is shown in Figure 2. While there was no statistically significant difference between the distributions, the biology focused course stands out as the only course where students were more likely to agree with the statement **after** rather than before taking the course. (On a 5 point Likert Scale where "Strongly Agree" was coded as 1, Biology had a mean of 1.5 pre-course and 1.32 post course, compared to Law with 1.42 and 1.54, and Media Computation with 1.06 and 1.33.)

As shown in Figure 3, as of a semester after the course was offered, 74% of the biology section has declared a CS major, and 52% took CS 2 in the next semester, as compared to 58% and 64% from the law section, and 85% and 82% in the media section. It is worth noting that quite a few students at our school always choose to defer CS 2 from the spring to the summer session.

4.4.3 Student Self-Efficacy. Students in the biology section were more likely to agree with statements like "I like writing computer programs" (shown in figure 4, mean of 1.63 versus 2.29 in law and 2.083 in media computation), "I think I could handle more difficult programming problems" (mean of 1.74 versus 2.04 in law and 2.00 in media computation), and "I am sure that I can learn programming" (mean of 1.26 versus 1.39 in law and 1.42 in media computation.)

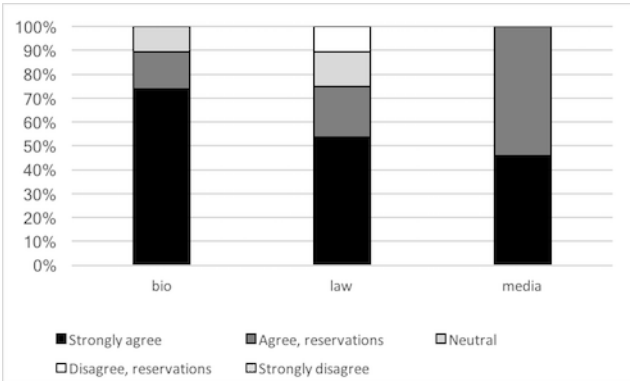

Figure 4: Responses to the statement “I like writing computer programs".

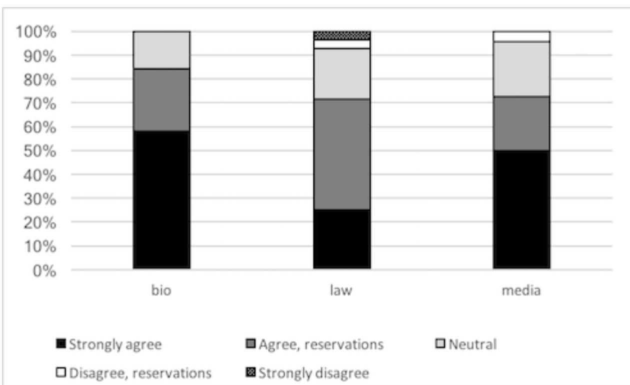

Figure 5: Responses to the statement “I would recommend this course to a friend".

While these results were not statistically significant, we take them as a promising sign of student self-efficacy.

4.4.4 Student Enjoyment of the Course. Students in the biology section were more likely to agree with the statement "I would recommend this course to a friend", as shown in Figure 5, with a mean of 1.74 on a five point Likert scale, versus 2.5 and 2.14 in the other two sections. Although that difference was not statistically significant, this is a promising sign that students enjoyed the course.

5 CONCLUSION

A CS + Biology approach to CS1 can be beneficial for both computer science and biology students. In this paper, we described the results of our initial offering of a Biology-themed CS1 course, created by very heavily modifying Harvey Mudd's CS 5 Green, to meet the needs of students at a large public university. We attracted a diverse set of students, including some that would not otherwise have taken a CS1 course. Students did just as well as traditional CS1 students on comparable questions on the final exam, indicating similar knowledge of CS content. Students who took the course report enjoyment of the course, and computer science in general.

6 ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant No. DUE-1612113.

REFERENCES

[1] Christine Alvarado, Zachary Dodds, and Ran Libeskind-Hadas. 2012. Increasing Women's Participation in Computing at Harvey Mudd College. *Inroads* 3, 4 (2012), 55–64.
[2] American Association for the Advancement of Science. 2011. Vision and Change in Undergraduate Biology Education: A Call to Action. (2011). http://visionandchange.org/finalreport/
[3] William Bialek and David Botstein. 2004. Introductory science and mathematics education for 21st-century biologists. *Science* 303, 5659 (2004), 788–790.
[4] Jane E. Caldwell. 2007. Clickers in the Large Classroom: Current Research and Best-Practice Tips. *CBE Life Sciences Education* 6, 1 (March 2007), 9–20.
[5] K. Clark, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and E. W. Sayers. 2016. GenBank. *Nucleic Acids Res.* 44, D1 (Jan 2016), 67–72.
[6] Catherine H. Crouch and Eric Mazur. 2001. Peer Instruction: Ten Years of Experience and Results. *American Journal of Physics* 69, 9 (2001), 970–77.
[7] Quintin Cutts, Angela Carbone, and Kelsey van Haaster. 2004. Using an Electronic Voting System to Promote Active Reflection on Coursework Feedback. In *Proceedings of ICCE 2004*. APSCE.
[8] Zachary Dodds, Ran Libeskind-Hadas, and Eliot Bush. 2010. 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)*. 219–223. https://doi.org/10.1145/1822090.1822152
[9] Zachary Dodds, Ran Libeskind-Hadas, and Eliot Bush. 2012. Bio1 as CS1: Evaluating a Crossdisciplinary CS Context. In *Proceedings of the 17th Annual conference on Innovation and Technology in Computer Science Education (ITiCSE)*. ACM, 268–272. https://doi.org/10.1145/2325296.2325360
[10] Harvey Mudd College. 2015. Common Data Set. Online at https://www.hmc.edu/institutional-research/institutional-statistics/common-data-set/. (2015).
[11] Sharon K Lanning, Sonya L Ranson, and Rita M Willett. 2008. Communication skills instruction utilizing interdisciplinary peer teachers: program development and student perceptions. *J Dent Educ* 72, 2 (Feb 2008), 172–182.
[12] Ran Libeskind-Hadas and Eliot Bush. 2014. *Computing for biologists: Python programming and principles.* Cambridge University Press.
[13] Roy P. Pargas and Dhaval M. Shah. 2006. Things are Clicking in Computer Science Courses. In *Proceedings of the 37th ACM Technical Symposium on Computer Science Education (SIGCSE)*. ACM, 474–78. https://doi.org/10.1145/1124706.1121489
[14] Pavel Pevzner and Ron Shamir. 2009. Computing Has Changed Biology—Biology Education Must Catch Up. *Science* 325, 5940 (2009), 541–542.
[15] Leo Porter, Cynthia Bailey Lee, Beth Simon, Quintin Cutts, and Daniel Zingaro. 2011. Experience Report: A Multi-classroom Report on the Value of Peer Instruction. In *Proceedings of the 16th Annual Confernece on Innovation and Technology in Computer Science Education (ITiCSE)*. ACM, 138–142. https://doi.org/10.1145/1999747.1999788
[16] Leo Porter, Cynthia Bailey Lee, Beth Simon, and Daniel Zingaro. 2011. Peer Instruction: Do Students Really Learn from Peer Discussion in Computing?. In *Proceedings of the Annual ACM International Computing Education Research (ICER)*, Michael E. Caspersen, Alison Clear, and Kathryn Sanders (Eds.). ACM, 45–52. Online at http://dl.acm.org/citation.cfm?id=2016911.2016923.
[17] U.S. News & World Reports. [n. d.]. Top 100 - Lowest Acceptance Rates. ([n. d.]). Retrieved August 31, 2017 from https://www.usnews.com/best-colleges/rankings/lowest-acceptance-rate Fall 2015 Acceptance Rate when last accessed.
[18] Beth Simon, Michael Kohanfars, Jeff Lee, Karen Tamayo, and Quintin Cutts. 2010. Experience Report: Peer Instruction in Introductory Computing. In *Proceedings of the ACM Technical Symposium on Computer Science Education (SIGCSE)*. 341–45. http://dl.acm.org/citation.cfm?id=1734263.1734381
[19] Robert H. Sloan, Cynthia Taylor, and Richard Warner. 2017. Initial Experiences with a CS + Law Introduction to Computer Science (CS 1). In *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '17)*. 40–45. https://doi.org/10.1145/3059009.3059029
[20] University of California at San Diego, Saint Petersburg Academic University, and the Russian Academy of Sciences. [n. d.]. Rosalind, a platform for learning bioinformatics through problem solving. http://rosalind.info. ([n. d.]). http://rosalind.info/
[21] Daniel Zingaro. 2010. Experience Report: Peer Instruction in Remedial Computer Science. In *Proceedings of World Conference on Educational Multimedia, Hypermedia & Telecommunications(Ed-Media)*, Jan Herrinton and Bill Hunger (Eds.). AACE, 5030–35. Online at http://editlib.org/p/36184/.
[22] Emile Zuckerkandl, Richard T Jones, and Linus Pauling. 1960. A comparison of animal hemoglobins by tryptic peptide pattern analysis. *Proceedings of the National Academy of Sciences* 46, 10 (1960), 1349–1360.
[23] Stuart Zweben and Betsy Bizot. 2017. 2016 Taulbee Survey. *Computing Research News* 29, 5 (2017). http://cra.org/crn/wp-content/uploads/sites/7/2017/05/2016-Taulbee-Survey.pdf