2017-18 Annual Report on
COMM ST 149: Computing Everywhere
Computational Literacy Initiative

Course Coordinator
Jeremy Birnholtz

Instructors
Elham Beheshti, Scott Cambo, Mike DeVito, Mark Diaz, Jes Feuston, Jeremy Foote, Jamie Gorson, Matthew Heston, Jim Maddock, CJ McFate, Sneha Narayan, Irina Rabkina, Daniel Trielli, Emily Wang

With Thanks for Administrative Assistance to
Madeleine Agaton, Amanda Davis, Jordan Dues, Colleen Jajewski, Rebekah Sigman, Deb Webster, Brad Zakarin

Introduction

The goal of Computing Everywhere (CE) is to make computational literacy more broadly accessible, particularly to a non-technical audience. Building on our initial offering of one section in 2016, and 4 in 2017, this year we again offered 4 sections (2 sections per quarter in winter and spring 2017) of a zero-credit (graded S/U) course consisting of a series of 5 weekly 2-hour workshops about key components of computational literacy. Based on student feedback and last year's action plan, we also added two one-day workshops this year in fall and spring. Overall enrollment decreased in CE programming from last year, with 40 students completing either the winter (14) or spring (12) workshop series, or one of the one-day workshops (10 fall; 4 spring). This compares with 49 students in 2016-17, without the one-day workshops.

Fourteen graduate students in the School of Communication and related programs proposed workshop topics and served as paid instructors and peer mentors, with faculty coordination and mentoring by Jeremy Birnholtz. Support was provided by the School of Communication, a curriculum development grant from the Alumnae of Northwestern, and an endowed gift supporting education in digital media.

Winter topics included: 1) basic fundamentals of coding in Python using the Earsketch package, 2) web structure and HTML/CSS, 3) cognition and computation, 4) the social implications of algorithms and 5) data and computational journalism.

Spring topics included: 1) public relevance algorithms and accountability, 2) web page accessibility and HTML/CSS basics, 3) the structures of online networks, 4) basics of artificial intelligence, and 5) network structures and mechanisms behind Google's search algorithms using the NetLogo package.

The topics for the two one-day workshops were: 1) learning to code through open-source projects (fall) and 2) web programming, with HTML, CSS and Javascript (spring).

Feedback was solicited from students each week via an anonymous questionnaire. Overall response from students was enthusiastic. This document provides a summary of our experience and lessons learned.

How It Worked

As in the past, our goal was to create a lightweight, low-pressure, hands-on and low-risk environment in which students could experiment with technology and develop computational literacy. Again the course carried zero credits and was a self-contained (i.e., without homework or reading assignments) series of five weekly 2-hour workshops. This allowed us to present the course to the students as a low-risk, relatively low-effort endeavor (see feedback below) that would be completed in the first half of the quarter, before exam and project deadlines kicked in. To make things more convenient and less formal for students, this year all sessions were held in residential Engagement Center classrooms located in campus residence halls.

We offered two 20-student sections of the course in Winter and Spring quarters of 2017, with a separate slate of workshops for each quarter that were repeated for the two sections. That is, there were 5 winter workshops that were each offered twice and 5 spring workshops that were each offered twice. To avoid conflicts with other courses, both sections were in the evening: 7 - 9 pm on Tuesday and Wednesday. Based on consultation with students, the one-day workshops were offered from 11 am - 3 pm on Sundays, with pizza and salad provided for a short lunch break.

In addition to these workshops for undergraduates, one workshop on the social implications of algorithms was offered as part of the CommFest alumni event in April, 2018.

Doctoral students and postdocs in Technology and Social Behavior, Media, Technology and Society, and related areas (e.g., HCI-focused students in Learning Sciences, Computer Science and Cognitive Science) were invited in the fall to submit proposals (individually or in pairs) in fall quarter 2017 to develop a single two-hour workshop (or a 4-hour workshop) and teach it twice. Compensation of $700 per instructor was provided (with the per-instructor amount the same whether there were 1 or 2 instructors).

A total of 13 proposals (including the one-time workshops) were submitted and received, all of which were evaluated by the course coordinator in terms of appropriateness, instructor experience and to achieve a suitable range of topics in each quarter. Ten proposals were ultimately selected for the workshop series, along with the two one-time workshop proposals. All applicants received detailed feedback on their proposals, and selected instructors were

assigned to peer-mentoring pairs to provide feedback on each other's proposals and emerging lesson plans.

Once workshop proposals had been accepted, a schedule of workshops was proposed (and modified as necessary), and instructors were asked to begin work on lesson plans and meet with the course coordinator. For Winter Quarter workshops, mentoring meetings occurred in late fall. For Spring Quarter workshops, mentoring meetings occurred in late winter.

Each workshop consisted of a combination of hands-on exercises, experimentation with coding and/or design, and group discussion led by the instructors. Instructors were encouraged, through the proposal instructions, peer mentoring and 1-2 mentorship meetings, to keep their workshop tailored to the School of Communication student audience. That is, concepts were expressed in terms familiar to Communication students and using familiar examples and metaphors. Hands-on experimentation with technologies and code wherever possible were strongly encouraged.

All instructors received detailed instruction feedback from the course coordinator (or, in a few cases, from a trained observer affiliated with the Searle Center for Advancing Learning and Teaching), who attended one instance of each workshop. Students completed a brief evaluation questionnaire following each workshop.

To manage the logistics of the course, workshop instructors were assigned basic administrative roles such as collecting student feedback, handling enrollment/registration questions, preparing the syllabus and course launch, etc.

Student Experience and Learning

In general student experience with the workshops was very positive. As Figure 1 shows, students generally felt they learned a lot and that the quality of instruction was good, and indicated that they wanted to learn more about the topics presented and felt all workshops should be offered again in the future. Qualitative comments indicated that students enjoyed the hands-on exercises, experimenting with different software tools and learning how computing systems work. In general, they were particularly enthusiastic about hands-on activities and challenges to solve problems on their own.

For example, one student in the Winter EarSketch session said, "Very fun and great way to learn basic ways to code." A student in the Winter HTML session said, "It was enlightening and also experiential, as we got to use the knowledge we acquired immediately."

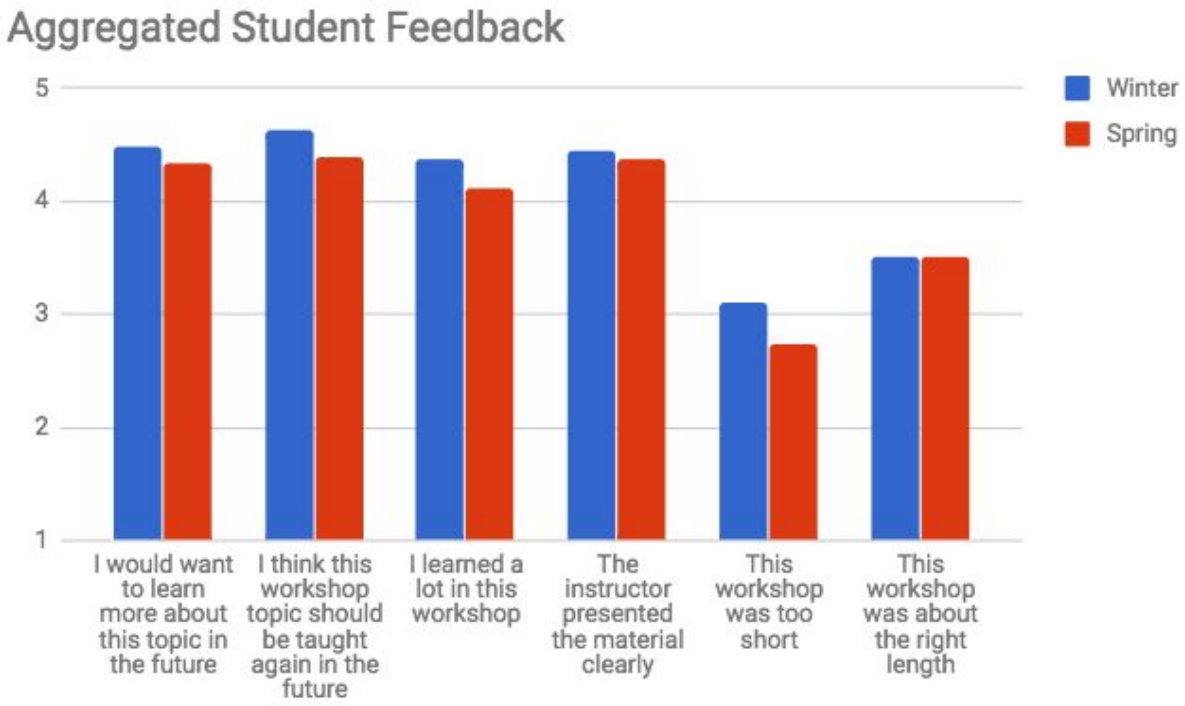

Figure 1: Averaged student feedback for all workshops, by quarter.
(1 = Strongly Disagree, 5 = Strongly Agree)

In general, students also appreciated the quality of instruction, noting in particular the patience, helpfulness and good nature of the instructors for the sessions. As in the past, Some students also indicated a desire for more depth in covering certain topics. For example, a student in the one-time HTML workshop in the spring wished there had been time to cover more commands. In general, one theme in student comments this year appears at first glance to be that most workshops did not try to take on an overly ambitious amount of the material. That is, one way to read these comments is to suggest that the students appetites were whetted, and they were ready for more.

Instructor Experience

Workshop instructors were asked to complete a brief questionnaire at the completion of each quarter, though instructors who taught during both quarters were not required to complete it twice. The questionnaire was completed by 7 (of 11) instructors. Feedback was generally positive, with some constructive suggestions for improving the experience.

Overall, 6 instructors said that their experience was “excellent” (3) or “very good” (3), with 1 indicating a “good” experience and 0 reporting a fair or poor experience. Moreover, all of the

instructors expressed at least some interest in teaching Computing Everywhere workshops in the future.

In qualitative comments, instructors appreciated the enthusiasm of the students, mentoring and feedback provided by the course coordinator throughout the course, and the experience of watching students learn. One instructor said, “I've enjoyed teaching the same material for multiple years, and would encourage other instructors to do so as well. Our material and presentation has changed quite a bit based on past experiences and feedback.” Another said “I liked that the students gave us full attention and were interested in the subject. We didn't have to do much in terms of motivating the class to learn.”

Instructors also raised several constructive issues to work on in future offerings, which are addressed in detail below. In general, the main points of constructive criticism were centered around (the seeming lack of) coordination among instructors (which is a repeated point from last year), and poor communication about enrollment numbers. In authoring this report, the coordinator is adding enrollment numbers themselves to this list of issues to address.

Lessons Learned and Goals for 2017-18

While the overall trajectory of Computing Everywhere was quite positive this year in terms of quality of instruction, expanding the offerings and maintaining momentum for the program, there are several challenges that we hope to address in the coming year and beyond. This section is a summary of those issues, some possible solutions and some preliminary proposals for addressing them.

Student Enrollment

In last year’s report student attrition was a significant issue, arguably due to the lightweight, zero-credit structure of the course. In 2016-17 year all of the sections of the course filled to capacity (20), but around 30-40% of students eventually dropped the course. This past year, attrition was less of a problem, but far fewer students enrolled in the course in the first place. It is not clear why enrollment numbers dropped, but this could be in part related to appetite saturation after a second full year (i.e., there was a pent-up desire for the course that led to enrollment spikes in 2016-17 that has since been sated). While it’s possible that this could be due to negative student experiences, this seems unlikely given the positive student feedback (albeit with some sample bias, as only students who enroll complete the feedback). It is also possible that we simply need to do a better job of getting the word out to students about the course.

Proposed 2018-19 action: I propose that we keep an eye on enrollment numbers and try to do a better job of getting the word out about the course in 2018-19. One possible strategy for this would be to hire an undergraduate to assist with course outreach and publicity, as well as getting an informal sense of student interests. If this year’s downward trends persist, more substantial corrective action should be considered in 2019-20.

Another possibility to consider would be to make CE a more structured part of the curriculum, with, for example, a computing literacy requirement for the major and/or modules, with CE being one of a menu of options for completing that requirement. This is currently being considered by the module faculty as a possibility.

Instructor Coordination

Several instructors felt that the decentralized approach to the course was perhaps too decentralized in that there was often little awareness among the instructors of who else was teaching and what they were covering. This was also a comment last year and, with apologies, the coordinator recognizes that he perhaps did not adequately address the concerns this past year.

In this regard, three suggestions appear particularly helpful:

A quarterly kickoff meeting was suggested in 2016-17, where instructors can meet each other, sync on course launch logistics and administrative jobs, and briefly summarize their workshops and identify potential synergies. This appears to be a very reasonable solution to the problem, though to be of maximum value to all instructors (including those who teach earlier in the quarter), the meeting should likely be held at the end of the previous quarter. This suggestion was not implemented in 2017-18, but it should have been.

Some concerns were also raised around the assignment of administrative tasks to the students, which were sometimes inconsistent across quarters and also sometimes resulted in disparate workloads or poor coordination. The timelines for these tasks were also unclear, and a concrete timeline has been very helpfully suggested by one of the 2017-18 instructors.

Proposed 2018-19 action: An instructor kickoff meeting will be held, either for the year or for each quarter. In addition, a clearer set of administrative descriptions and a concrete timeline will be developed. These will be available in the fall quarter when proposals are solicited and tasks are assigned for the year.

Teaching Coding Fundamentals

This issue was not raised in 2017-18, but it did come up in 2016-17 and is included here because it was not addressed well in 2017-18. The coordinator feels it merits further discussion.

In 2016-17 instructor feedback suggested that students want to learn more about coding, but that it was not always clear how the fundamentals of coding were being taught. During some workshops some fundamental concepts were explicitly covered, but in other cases they were not. There is also little agreement on what constitutes ‘fundamentals’ and the depth in which these concepts should be covered in a lightweight workshop series.

This is a complicated problem with many possible approaches and few ideal solutions. Teaching 1-2 workshops focused only on coding fundamentals is a possible approach, but this represents a substantial fraction of any given quarter of instruction and some have expressed concern that this might feel too much like a 'class' instead of very applied, hands-on workshops with cool concepts.

Teaching an in-depth workshop that lasts a day or more and focuses on coding fundamentals is also a viable solution, but requiring this is problematic in that it would likely detract significantly from enrollment. Unless this is deemed absolutely necessary, this would probably be in conflict with the goal of making the course widely accessible and lightweight. It would be entirely possible, however, to offer an optional in-depth workshop.

A hybrid approach (experimented with in an ad hoc way in Spring 2017) would be to have instructors teach some elements of coding fundamentals in each workshop, but mention that students who want to learn more about this should take the in-depth workshop and/or look at online resources that are available. This would involve some agreement on what these concepts are and could involve workshop proposals that explicitly identify a concept that the workshop would address.

Previously proposed 2017-18 action: A document will be set up for identification, definition and discussion among past instructors of core concepts that should be addressed in every quarter of Computing Everywhere. Concepts should be identified and defined ***before*** *the call for workshop proposals is released in the fall. Each proposal will be required to identify one of these concepts to address (with the potential for allowing a rationale in a case where a proposal does not), and proposers will be encouraged to coordinate on what they are thinking of offering (either via, e.g., Slack or a Google Doc). For more depth, instructors will be encouraged to propose an in-depth coding fundamentals workshop (see below) to be offered sometime in 2017-18.*

Reflection after 2017-18: While this Google doc was set up and contributed to, the requirement of identifying a computational concept within each workshop proposal was largely ignored by proposers and not enforced by the coordinator. At least on the coordinator's part, this was not an inadvertent oversight. It became clear in the discussions within the Google doc that all of the issues mentioned above (i.e., lack of agreement on coding fundamentals) made this a less than ideal requirement. It may be that the lightweight, decentralized Computing Everywhere framework is simply not a good one for teaching coding fundamentals and a separate course and/or structure are needed. Additional feedback and discussion of this issue are most welcome, but no concrete action is currently proposed for 2018-19.

Depth of Instruction and Mentoring

As in 2016-17, student and instructor feedback suggests a desire for more depth on particular topics, both via "deeper-dive" workshops and perhaps additional online instructional materials.

In 2017-18 we adopted the in-depth workshop format. This resulted in two workshops being offered that were generally successful and well-received by the students. While the format and content appeared to work well, there were two challenges:

1) the lack of a structured proposal deadline process appeared to result in a small number of unsolicited proposals. The accepted proposals were both solicited by the course coordinator. It could be that a deadline is needed to encourage proposal submission and/or that it may simply be a good mechanism to ask existing instructors to offer in-depth workshops on particular topics. Feedback on this is welcome.

2) the highly sporadic nature of these programs meant that it was hard to build momentum and attract student enrollment/attention. This is essentially a harder version of the “getting the word out” problem described above for enrollment, and also requires more effort on promoting the workshops.

Proposed 2018-19 action: As in 2017-18, anybody is welcome to submit proposals for one-time workshops at any time, but a small number will be solicited from existing instructors to ensure that some workshops are offered. We will aim to offer 1-2 workshops per quarter in 2018-19.