Computing for Medicine: Can We Prepare Medical Students for the Future?

Marcus Law, MD, MBA, MEd, CCFP, FCFP, Paula Veinot, MHSc, Jennifer Campbell, MMath, Michelle Craig, MSc, and Maria Mylopoulos, PhD

Abstract

Problem

Technology can transform health care; future physicians need to keep pace to ensure optimal patient care. Because future doctors are poorly prepared in computer literacy, the authors designed a computer programming certificate course. This Innovation Report describes the course and findings from a qualitative study to understand the ways it prepares medical students to use computing science and technology in medicine.

Approach

The 14-month Computing for Medicine certificate course (C4M, offered beginning in February 2016), University of Toronto, is comprised of hands-on workshops to introduce programming accompanied by homework exercises, seminars by computer science experts on the application of programming to medicine, and coding projects. Using purposive and maximal variation sampling, 17 students who completed the course were interviewed from April–May 2017. Thematic analysis was performed using an iterative constant comparison approach.

Outcomes

Participants praised the C4M as an opportunity to achieve computer literacy—including language, syntax, and fundamental computational ideas (and their application to medicine)—and acquire or strengthen algorithmic and logical thinking skills for approaching problems. They highlighted that the course illustrated linkages between computer science and medicine. Participants acknowledged a sometimes-existent chasm between producers and users of technology in medicine, recommending two-way communication between the disciplines when developing technology for use in medicine.

Next Steps

We recommend that medical schools consider computer literacy an essential skill to foster future collaborative computing partnerships for improved technology use by physicians and optimal patient care. We encourage further evaluation of future iterations of the C4M and similar courses.

Problem

In the last few years, a number of medical schools have redesigned their curricula to add competencies and learning objectives around the sophisticated use of information technology, labeling it in a variety of ways, such as health informatics,[1] biomedical informatics,[2] or medical informatics.[3] Although these efforts teach medical students about the use of technology in their future practices, they do not teach students how to write computer programs. Learn-to-code programs for medical schools are largely absent from medical school curricula. Given the lack of preparation of future doctors in computer literacy suggested by this omission, we designed a hands-on experience for students to learn basic computer programming skills, including how to code.

Our contention is that without such cross-domain understanding, computer science and medicine will continue to work in silos and that problems with technology adoption (e.g., poor understanding of value, lack of utility) will be perpetuated even though future physicians should be compelled to keep pace with technological advances, which can transform health care, to ensure optimal patient care.

The purpose of this Innovation Report is to briefly describe our elective computing course and discuss the findings from a qualitative study to understand how the course prepares medical students to use computing science and technology in medicine.

Approach

We developed, and in February 2016 began offering, a 14-month Computing for Medicine certificate course (C4M), for students enrolled in the University of Toronto MD program. The course was composed of three core phases: (1) a series of hands-on workshops to introduce programming (students with minimal prior programming experience received an additional introductory session) accompanied by homework exercises; (2) consolidation workshops to provide further instruction and practice; and (3) seminars delivered by experts from various computer science fields, who discussed the application of programming to medicine accompanied by coding projects that corresponded with each seminar. The course was a collaboration between the University of Toronto Faculty of Medicine and Department of Computer Science and was taught by faculty from the Department of Computer Science.

Description of the C4M

We assigned students a skill level based on their self-reported prior programming experience.

Course details are as follows: In phase I, principles of programming, students participated in a series of programming

Please see the end of this article for information about the authors.

Correspondence should be addressed to Marcus Law, Faculty of Medicine, University of Toronto, Medical Sciences Building, 1 King's College Cr., Room 3157, Toronto, ON M5S 1A8, Canada; telephone: (416) 978-4543; e-mail: marcus.law@utoronto.ca.

Acad Med. 2019;94:353–357.
First published online November 13, 2018
doi: 10.1097/ACM.0000000000002521

workshops (three to four sessions depending on skill level over a 3-month period), which included didactic teaching, demonstrations, and smaller in-class practice exercises. In this phase, students were also required to complete larger coding exercises as homework before the next workshop, which was at least 2 weeks later. As mentioned above, students with minimal prior programming experience also completed an additional introductory session during this phase. The objectives of this phase were to enable students to write very basic Python programs; trace basic Python programs involving lists, dictionaries, and files; and recognize good practices in software design.

In phase II, consolidation (five sessions over a 4-month period), students attended workshops with a focus on the consolidation of learning over time. In this phase, they were required to complete two large medical-themed Python projects that involved nearly all of the programming concepts they had learned in phase I. The objectives of this phase were to enable students to write programs that combined concepts from phase I, write programs to solve a problem, use good practice in software design consistently, and use a debugger to find mistakes in a program written by another author.

Students progressed to phase III, enrichment (six sessions over a 7-month period), after having gained experience solving Python programming problems in phases I and II. In this phase, they enhanced their understanding of how computing can be applied to medicine through six 2-hour seminars conducted by experts in an area of computing related to medicine and computing science faculty who taught additional relevant programming or computer science concepts. Students were required to complete projects related to three of the seminars to demonstrate the application of knowledge and skills learned. There were a variety of project options available to encompass the full range of learning needs (from novice to more advanced learners), and students self-selected the projects they wanted to work on.

Course evaluations

We conducted a qualitative evaluation (ethics approval was received from the University of Toronto Research Ethics Board) to explore participants' experiences and perceptions of the C4M. We applied purposive and maximal variation sampling (i.e., sampling based on prior programming experience and male/female) to recruit 17 students who had completed the course to participate in interviews. Interviewees received a $20 gift card for taking part in the study. On obtaining written consent, a research assistant with qualitative research experience (P.V.) conducted one-on-one in-person or phone interviews over a 1-month period (April–May 2017) using a semistructured interview guide (Appendix 1). Interviews were conducted to obtain theoretical saturation[4] (i.e., until no new ideas were presented and participants' evaluation of the course was well understood). Interviews ranged from 26 to 61 minutes (mean = 39 minutes, median = 34 minutes) and were audio-recorded and transcribed verbatim. Immediately following each interview, P.V. also documented field notes.

We conducted a thematic analysis[5] of the transcripts using an iterative constant comparison approach, which emphasizes an inductive and open approach to data collection, allowing understanding to emerge through careful analysis of the data.[4] Data were coded, compared with each other, and grouped into themes. An iterative approach to data collection helped to refine the interview questions and develop more targeted questioning. This continuous refinement of the research process yielded a substantive understanding of participants' experiences and perceptions of the course.

Outcomes

We classified data into two main themes: value of the course (with the subthemes of benefits and improvements) and potential application of learning (with the subthemes of current or future application and facilitating communication and a dialogue between the two disciplines). See Table 1 for example quotations.

Value of the course

Benefits. All participants highlighted the value of the C4M and reported that they liked the structure of the course, enjoyed the assignments and projects, and achieved their learning goals. Some students noted that their confidence in programming improved. Although some participants experienced frustration and felt overwhelmed at times (e.g., they—especially novice learners—were challenged by learning new concepts, struggled with finding time to practice, and had difficulty completing assignments), even these individuals stated that the structure was appropriate for enabling students to succeed. Most participants reported that the C4M provided them with a formal opportunity to learn computer programming skills as a legitimate part of their medical training, affording them an understanding of computer science and the nomenclature associated with the discipline. Participants generally attested that the course was well organized and that the facilitators were supportive, were approachable, and responded to questions in a timely fashion.

Aside from language and syntax, most participants described gaining a familiarization with fundamental computational ideas (e.g., understanding how machine learning can solve real-world problems). Many also reported acquiring or strengthening algorithmic and logical thinking skills. These skills expanded the ways that students thought about and approached problems. All participants reported that the C4M opened their eyes to the applicability of fundamental computational ideas to medicine and illustrated the linkages between the disciplines of medicine and computer science.

Improvements. In addition to the benefits of the C4M, participants also recommended some improvements. Prominent suggestions included having more in-person help (e.g., for students at the satellite site, who were video conferenced into the course, and more office hours); finding ways to better tailor or customize the workshops to meet the learning needs of all learner levels from novice to experienced (i.e., some found it to be challenging, others found it to be too superficial, while others thought it was a good refresher); having experts who have a crossover role (clinical and computer science/technology) to present examples of projects that the average physician could reasonably integrate into their practice; expanding the C4M to the whole of the medical school, even at an introductory level, to spark greater

Table 1

Example Quotations, From Semistructured Interviews Exploring Participants' (n = 17) Experiences and Perceptions of the Computing for Medicine Course, University of Toronto, April–May 2017

Theme	Example quotations (participant number)[a]
Value of the course	
Benefits	• I thought it was a very good structure, especially for someone like me, who came in with some knowledge base already. (#01)
	• The course itself I think is fantastic, and I hope that you continue to [do] something like this, and I'm also really happy that this was done in concert with the computer science program. […] I'm personally happy that I took part in it, and I'm glad that even though it was lot of time to spend to kind of figure things out, I think it ultimately was very worth it. (#05)
	• So, I think the biggest benefit for me was probably the fact that it really solidified the fact that I want to do this down the road. It made it clear to me that this is something I really want and need to have in my life going forward, and I enjoy it. (#16)
	• I just appreciated, having informally taught myself how to program, I appreciated how the course […] approached it more formally, where it actually taught me how to think like a computer scientist, as opposed to somebody that was just picking up the language on their spare time. It made the art of computer programming more systematic and deliberate than what I had been used to. I would just learn whatever came up and I wasn't really doing things in order. But the course is really good for introducing me to the discipline of computer science itself. (#13)
	• That was really good, I definitely think about problems a little bit differently now, in terms of really … this idea of deconstructing your normal human way of solving problems, and really thinking about what do I actually do when I want to solve a problem, and then being able to break that down into lines of code. I do think … I can actually identify problems in the real world now that I think I'd be like, "Oh, this is a problem that could easily be done with a simple script. Or this is a problem that we could solve with machine learning or something like that." I really like having that skill set. (#03)
	• I think it was actually very beneficial for learners, especially me as kind of a young learner, in terms of not having had much experience. The boot camp really helped expose me to the computer programming and start to learn all the basics. […] Initially, I didn't have much exposure to computer programming, so I guess I wasn't aware of all the different applications that computer programming and technology can have in the medical world. (#07)
Improvements	• But I think some sort of in-person help would be useful to the students in the [satellite site], just sort of as an aside to what I was mentioning earlier about the advantages and disadvantages of video conferencing. (#12)
	• I think more office hours would have helped in that case, having more access to somebody in person, because a lot of the stuff was troubleshooted over e-mails. (#17)
	• In the boot camp part where they were teaching the skills, like sometimes I found it going too slow because of some other people who maybe didn't have as much experience. But then other parts I found like they were going too fast. Other people wanted it to go faster, but it was too fast for me. (#02)
	• I think there is certainly room to grow in terms of understanding how to build a course that does what it says on the box. Because I think in certain ways it was very difficult for my peers who have never worked with computers before. I felt many of them weren't actually able to finish the course. (#05)
	• It would have been useful to see a model of where we could end up […] I think there was one clinician who was using computer science, but the majority weren't. I think it would have been useful to see that. (#08)
	• […] computer scientists coming in with just solutions and a physician who is also a computer scientist or who has an interest, I think all three perspectives are very valid. (#11)
	• I actually do think it would be very beneficial for students, and I think all students should take it and have it maybe be incorporated into the medical program. (#07)
	• I know that the course originally planned to have a practicum component in the summer, which I guess that didn't go through, but I think that would have been really helpful. Because I think most people don't take the effort to actually go out and try and apply the skills. I could see, for a lot of people, they'll do this, they'll be like, "Oh, this was cool to learn," but they may never visit it again. (#03)

(*Table continues*)

interest in this field; and including a practicum opportunity.

Potential application of learning

Current or future application. All participants indicated that they saw themselves using or would like to apply what they had learned in the C4M at some point in their careers. Some participants had already applied their learnings to research projects and summer jobs. Even participants whose interest in computer programming preceded enrollment recognized that the C4M enhanced or reinforced their interest and solidified some of their skills and thinking about how to combine these two fields in a career as well as the paths they might take to accomplish this.

Facilitating communication and a dialogue between the two disciplines. Participants acknowledged a sometimes-present disconnect between the producers (computer scientists) and users (physicians) of technology in medicine. They suggested that part of the solution was to ensure two-way communication between the disciplines of computer science and medicine when developing technology that is meant to be adopted in medicine. Many participants described themselves as wanting to be part of this communication between the two disciplines. Participants articulated that promoting a dialogue between the two disciplines would ensure the development of realistic, usable, and effectively used technology.

Next Steps

Our findings suggest that students who participated in the C4M felt more prepared and motivated to promote a dialogue between the disciplines of computer science and medicine. Thus, in our experience, exposing students to computing science through a certificate program in the MD curriculum can reinforce a valuing and understanding of technology and encourage students to foster future collaborative computing partnerships. Given that technology is ubiquitous and changing rapidly, we recommend that medical schools consider computer literacy as an essential skill for future physicians. Teaching interested students computer literacy skills so that they have content knowledge and confidence will allow them to be

Table 1
(Continued)

Theme	Example quotations (participant number)[a]
Potential application of learning	
Current or future application	• I was doing research this summer, and I ended up using some of the skills to write a program that I used in my research to consolidate a bunch of patient information from hundreds of charts. So, I was able to develop a code for that which made me more efficient in the work I was doing at the hospital over the summer, so yeah that helped me. I did a lot of troubleshooting and trial and error, but I feel like the course helped me use some of the basic skills and build on those in order to achieve my goal. (#09) • Yes, I'm currently working on a start-up actually, and we're creating software solutions. So, I'm tackling the app development and the machine running aspects of that solution and working with a couple of computer programmers. So, they're also teaching me stuff, but it allows me to just continue what I've learned through the course and apply it to something that I'm doing personally. (#13) • The summer following the first year that I spent with Computing for Medicine, I took a research job doing general surgery at [hospital name redacted]. And one of the things I put on my resume was the fact that I had done, I was in a course, learning to compute, and was saying to them that if there was any avenue that I can apply what I'm learning in computing … if there is some sort of project that I can either crunch numbers, do some different program, some kind of computing intersection with this, then I am someone who has this relatively rare set of skills. (#14) • Again, I think I came into it thinking about a career choice that combined the two, so I think it gave me more ideas in terms of ways I can combine the two. I don't know if it changed completely the direction, but I think it gave me more fuel to support some of the interests and ideas that I have already been having, like makes them a little more concrete. (#04) • No, I think if anything it enhanced it. […] For example, it made me surer of pursuing specialties that are more focused on research and using technology to advance the care we offer patients currently. (#09) • I think I will want to be thinking of ways to integrate computers in a way that will facilitate my job. […] What is it that I'm not good at that I can have a computer do for me? […] I think I'm definitely inclined to something that's more directly involved with computer technology. I'm not sure necessarily whether the course changed that or more confirmed that for me, because of my background in it. The course enabled me to see more clearly computing in medicine and to think about what professions would be most suited to incorporating that in. That's why I'm thinking very much about radiology is because there's so many advances currently being made in technology. (#12)
Facilitating communication and a dialogue between the two disciplines	• I think with time I will become more of an expert in medicine than I will ever be in programming. There are people, engineers and computer scientists that know far, far more than I will ever know. What would be helpful is that I know the needs in medicine and I have a good understanding of the potential and the limitation and the ability of pro- … computer science. I can identify things that can be addressed with machines and programs and things that we can't solve with programs. I think I'll bring a more realistic view of what we can do and how we can use computers in medicine. (#10) • Eventually in the future when I have my own practice, I definitely want to be able maybe to work with engineers or computer scientists to build new applications for sure. I don't think you as a physician are going to be sitting there and coding anyway, but this course is also about letting me know what the technologies are and how. At least you know that you have a realistic view if something is doable or not because you have a little bit of background. (#11) • And so, having sort of a middle man between the clinician and the computer scientist, who understands something of both worlds, I think will facilitate communication and the collaboration between the computer scientist and the clinicians. … And so, thinking about my future role, I think it would be useful as sort of the intermediary between, as a clinician, when I become a clinician, I will have a much better sense of what are the needs and what is the perspective of clinicians on certain problems. But having still that little bit of awareness of what is the approach and the modus operandi of the computer sciences so that when you bring them in to involve them in your work, you'll be able to translate your clinical problem into something that they're able to understand. And once they understand it, they can help you fix it. (#12)

[a]"[…]" indicates that text has been omitted, while "…" indicates a pause in speech.

ambassadors who can encourage their physician colleagues to engage with technology. At the same time, having computing language skills and motivation will enable physicians to communicate more effectively with technology developers. It is anticipated that such cross-domain understanding would result in improved technology use by physicians striving toward optimal patient care.

Much like the physician quality improvement movement of the past 10 years, which has been providing physicians with training and support for engaging in quality improvement, an investment in physicians' confidence, knowledge, and skills in computing has the potential to similarly enhance the delivery of quality patient care. The question is: How this can be rolled out in a broader fashion? What should be elective, and what should be mandatory? Guzdial[6] argues that computing educators who are teaching elective courses such as ours need to develop the learning goals for their materials by considering the eventual communities of practice and motivations of the learners. In addition, it may be worthwhile to consider applicants' previous computing experiences and their interest in this topic area when making admissions decisions for these types of courses.

This Innovation Report explores the implementation of a computer programming certificate course for medical students at one Canadian medical school. Whether our data would hold true for other geographic contexts and whether making the course a mandatory one would yield different findings could be subjects for future research. Also, the participants we interviewed had self-selected into the course and completed it in its entirety. It is unknown whether students who did not fully complete it would have had similar views. We encourage further evaluation of future iterations of the C4M and similar courses to better understand learning transfer. Given the limitations of existing outcome-based evaluation models to capture the effects of health professions education,[7] we chose qualitative inquiry to help us understand the complexities of the educational intervention and thus generate information that we hope will be useful for curriculum designers.

Funding/Support: None reported.

Other disclosures: None reported.

Ethical approval: This project received ethics approval from the University of Toronto Research Ethics Board (protocol reference no. 34180).

M. Law is associate professor of family medicine and director, Foundations, MD Program, University of Toronto, and director of medical education, Michael Garron Hospital, Toronto, Ontario, Canada; ORCID: https://orcid.org/0000-0001-9746-6381.

P. Veinot is an independent research consultant, Toronto, Ontario, Canada.

J. Campbell is associate professor, Teaching Stream, Department of Computer Science, University of Toronto, Toronto, Ontario, Canada.

M. Craig is associate professor, Teaching Stream, Department of Computer Science, University of Toronto, Toronto, Ontario, Canada.

M. Mylopoulos is associate professor, Faculty of Medicine, and scientist, Wilson Centre, University of Toronto, Toronto, Ontario, Canada.

References

1 Hersh WR, Gorman PN, Biagioli FE, Mohan V, Gold JA, Mejicano GC. Beyond information retrieval and electronic health record use: Competencies in clinical informatics for medical education. Adv Med Educ Pract. 2014;5:205–212.
2 Silverman H, Cohen T, Fridsma D. The evolution of a novel biomedical informatics curriculum for medical students. Acad Med. 2012;87:84–90.
3 Burnette MH, De Groote SL, Dorsch JL. Medical informatics in the curriculum: Development and delivery of an online elective. J Med Libr Assoc. 2012;100:61–63.
4 Glaser BG, Strauss AL. The Discovery of Grounded Theory: Strategies for Qualitative Research. Chicago, IL: Aldine Publishing; 1967.
5 Braun V, Clarke V. Using thematic analysis in psychology. Qual Res Psychol. 2006;3:77–101.
6 Guzdial M. Learner-centered design of computing education: Research on computing for everyone. Synth Lect Hum Cent Inform. 2015;8:1–165.
7 Haji F, Morin MP, Parker K. Rethinking programme evaluation in health professions education: Beyond "did it work?" Med Educ. 2013;47:342–351.

Appendix 1

Semistructured Interview Guide, Used to Explore Participants' (n = 17) Experiences and Perceptions of the Computing for Medicine Course, University of Toronto, April–May 2017[a]

1. What were your original learning goals before you started this computing course?
   Probe: How did you self-select into the course?
2. Do you think that you achieved your learning goals by completing this computing course?
   Probe: How successful was the structure in helping students succeed?
3. Was there anything unexpected that you learned?
   Probes: Tell me more about it. Why are they unexpected?
4. Do you have any plans to further your learning in this field or similar field?
5. Do you feel this computing course changed the way you think as a medical student?
   Probe: E.g., better at abstract thinking b/c of learning coding skills?
6. Do you feel this computing course changed the way you provide clinical care?
7. Do you feel this computing course changed the way you learn as a medical student?
8. Do you feel this computing course changed the way you see yourself as a future physician?
9. Do you feel this computing course changed your future career choice and why?
   Probe: What are your plans in the next few years now that the course is over?
10. What do you think technology can do for you in your future practice?
    Probes: How can technology influence quality, patient outcomes and safety and value issues? How may you play a role in this? Did this course provide you with the skills to do this?
    a. Do you have any plans to incorporate what you learn or will learn in this field into your future career?
       Probes: How will you figure out how to use/apply what you've learned in the future? How will you get to that point? What potential issues may arise between physicians and computing science/technology people working together? What role do you see yourself having to help bridge the divide or gap between these two communities for future collaborations?
11. Do you have any other comments?

Abbreviation: b/c indicates because.
[a]Edited for format only.