A Curriculum Model for Preparing K-12 Computer Science Teachers

Cameron L. Fadjo[1], **Ted Brown**[2], **and Leigh Ann DeLyser**[3]
[1] The CUNY Institute for Software Design and Development,
City University of New York, New York, NY, United States
[2] Department of Computer Science, Graduate Center,
City University of New York, New York, NY, United States
[3] Department of Computer Science, New York University
New York, NY, United States

Abstract - *A comprehensive approach to preparing in-service and pre-service K-12 teachers to teach computer science and software engineering through an interdisciplinary masters-level graduate program is proposed and is described herein. Consisting of traditional graduate-level courses, field-based classroom experiences, and industry mentorships, the proposed program is designed to prepare current (in-service) or prospective (pre-service) teachers to develop the practical, theoretical, and conceptual knowledge necessary to effectively teach computer science and software engineering in the K-12 classroom. Current trends in computer science K-12 teacher education are examined and our innovative model for comprehensive teacher preparation in the fields of computer programming, computer science, and software engineering instruction is described.*

Keywords: Computer Science Education, Software Engineering Education, Computer Science Teachers, Computing, Computational Thinking, Pedagogy

1 Introduction

Computer science and software engineering education is re-emerging as a major academic discipline within the 21st century educational system. From online schools and MOOCs to immersion schools and programs [15], the traditional education system is being transformed, upended by a surge in interest fueled by industry and government mandates for innovative approaches to preparing students for careers in new and emerging high tech fields [1]. Paradoxically, however, the rapid rise in innovative programs to improve the education of young students quite frequently overlooks the professionals that are most likely to lead the transformation - the teachers. In the United States, where the need to prepare students for the new economy is particularly acute, there are a few programs, like the NSF-funded CS10K project [16] and the Georgia Computes! [17] program, that are working toward solving the most critical question in computer science education: How do we best prepare computer science educators to enter the classroom and teach computer science to all students [18]?

Since the post-dot-com resurgence of computer science, many institutions of higher education have attempted to address the need to prepare the two major teacher populations: namely, pre-service and in-service teachers. For pre-service teachers, those who are enrolled in an undergraduate or graduate-level teacher preparation program before entering the classroom as a full-time professional, the number of programs available in computer science education has slightly increased in tandem with the perceived local need to fulfill this occupational role. In Indiana, for example, Purdue University has developed a 'Computer Science Teaching Endorsement' which offers an undergraduate a sequence of education and computer science courses that would provide her or him with a 'teaching endorsement' [4] in K-12 computer science. To ensure that students in this program are 'prepared' to teach, endorsement candidates in this program take two courses in the CS Education focus area [4] ('Contemporary Issues in Computing,' an examination of technology and society, and 'Methods of Teaching Computer Science,' which surveys current theories of computer science education), as well as four CS content courses in computer programming and computer science [6]. Unfortunately, severely low enrollment numbers [7] and a lack of state-level recognition for computer science as an academic content area [8] have contributed to an ineffective start to this teacher preparation program in CS Education at Purdue. Remarkably, in states where high school-level computer science courses count as a "core" credit and programs have been developed at a local college or university to prepare the pre-service teachers, such as in Georgia, low enrollment numbers continues to be an issue for pre-service teachers [7]. In both cases, the lack of coordination between the colleges/universities and the state-level Boards of Education seem to be a primary contributor to the lack of success in launching rigorous computer science education programs that lead to a professional pathway for pre-service teachers.

For in-service teachers, professional development models in computer science education have often been designed to take a professional, either teacher or industry, and prepare them through a one-day, one-week, or longer workshop or class for the rigors of teaching computer science

at the high school level. These programs have been run through the College Board (with AP workshops), through universities or colleges, or even through individual school districts that service multiple schools and have a cadre of teachers to train. Programs for in-service teachers, such as the ones mentioned above, often focus on the individual needs of a particular population or school district and neglect to consider the need to build capacity within the immediate cohort to build capacity. With the graduate-level course sequence we are proposing, the bigger picture of developing a program that both meets the current needs of the computer science and software engineering programs throughout New York City [14] was the bedrock of our discussions to produce a model of professional development that is inclusive of pre-service and in-service teachers who may or may not have a background in computer science.

1.1 Certification and the Challenges It Presents

An important piece to consider when designing any pre-service or in-service teacher preparation program in the US is that of teacher certification. Teachers of computer science face many issues regarding certification, and the individuality of state certification programs present an additional challenge when considering the design and development of a national program for teacher preparation in computer science education [9]. Many states allow any subject certified teacher to teach computer science, while others require either a specific subject discipline such as Mathematics, Business/Library Media, Science or a technology specific certification, but have no certification for computer science.

In many of the current teacher preparation programs around the United States, teachers receive formal pedagogical training in the prototypical core discipline (such as Mathematics or 'Technology') but many, if not all, of these same programs that claim to prepare teachers for computer science neglect the integration of best practices from the field of computer science education into the instruction of computer science pedagogy. The 'Computer Science Education', Educational Specialist program at Nova Southeastern University [5], for example, is a graduate-level program that is designed to prepare teachers for the K-12 computer science classroom. Unfortunately, the program takes a modest approach to ensuring that a relatively broad range of STEM student are prepared to teach some form of computer science in the K-12 classroom and do so at the risk of neglecting the core of computer science education. Given the recent emergence of a core set of best practices that computer science educators use to teach abstractions [20, 21], there is also research to support the identification and use of pedagogical content knowledge (PCK) that is specific to the academic discipline being taught [10]. The lack of specific preparation in computer science pedagogy leaves many teachers with the need to do additional professional development outside of their programs in order to feel confident teaching computer science.

In New York State, computer science is most often counted as a mathematics credit, and most teachers who teach computer science also have a mathematics certification [9]. Preparing teachers for appropriate certification in mathematics education as well as computer science education are conflicting goals, however we outline a sequence of graduate-level courses that, we believe, must be integrated in any program addressing teacher preparation that ensures teachers are prepared for the computer science classroom while being simultaneously prepared to obtain a state-level teaching credential.

2 Building Teacher Capacity Through a Comprehensive Curricular Program

New York City is well positioned to create a large-scale computer science teacher educational program. Over the past couple of years, the Mayor of New York City, Michael Bloomberg and his administration have developed a number of initiatives and programs that are designed to establish New York City as a national high-tech hub. Many of these programs, such as the Center for Urban Science and Progress (CUSP) [11], and collaborations, such as the partnership between Cornell University, Technion – Israel Institute of Technology, and the City of New York, have become the hallmark of the public-private partnership model [12].

At the primary and secondary school level, in particular, the Bloomberg Administration has been working to ensure that computer science and software engineering education is open and accessible to a wide range of students. Last year the Bloomberg Administration, in collaboration with private industry partners such as venture capitalist Fred Wilson from Union Square Ventures and software engineers from high tech companies such as Google, Facebook, and Foursquare, partnered with the NYC Department of Education to create the Academy for Software Engineering (AFSE) [13], with a second school (the Bronx Academy for Software Engineering - BASE) opening in the fall of 2013. More recently the Administration announced the Software Engineering Pilot (SEP) program, a comprehensive, full-year computer science and software engineering curriculum, that will be placed in twenty NYC public schools this coming fall (2013) [14]. Forty in-service teachers, two from each of the twenty SEP schools, have been selected to participate in an extensive professional development program that will culminate in the teaching of a broad-based computing curriculum to over 2,000 students across NYC.

In the case of all three NYC Department of Education Software Engineering initiatives (AFSE, BASE, and SEP), there currently exists no graduate-level courses or programs, either locally or nationally, that prospective pre-service or in-service teachers can take to prepare them for a career as a computer science and software engineering educator. For AFSE and BASE the need to identify qualified educators is particularly acute since both schools require teacher candidates with a substantial background in computer

science, pedagogy, and instructional technology as a baseline qualification.

3 Proposed Computing Education Degree

A thirty-six credit masters degree has been mapped out for in-service and pre-service teacher candidates in conjunction with experts in computer science, the learning sciences, computer science education, and instructional technology from multiple departments and institutes within the City University of New York (CUNY) and leaders from the Software Engineering high schools (AFSE) within the New York City Department of Education. The following nine courses (27 credits of a 36 credit masters) are the core of the proposed graduate-level course sequence that will eventually culminate in a Master of Science (M.S.) degree in Computing Education (see Table 1).

Group	Course	Semester	Prerequisite(s)
Computing & Mathematics	Computational and Mathematical Thinking 1 (CMT 1)	A	-
	Computational and Mathematical Thinking 2 (CMT 2)	B, C	CMT 1
	Computational and Mathematical Thinking 3 (CMT 3)	C, D	CMT 2
	Discrete Structures (DS)	A	Calculus
	Algorithms (AL)	B, C	CMT 1; DS
	Computational Science (CN)	C, D	CMT 2; CN
Computing Education	Pedagogical Practice and Methods in Computation and Mathematics (PMCM)	A, B, C	CMT 1
	Practicum in Computing Education (PCE)	A, B, C, D	PMCM
Learning Sciences & Technology	Foundations: Learning Sciences & Technology (LST)	A, B, C, D	-

Table 1: Proposed Master's Degree Courses in Computing Education

Notes: The 'Group' corresponds to the specific discipline with which the courses are most directly connected. 'Semester' A, B, C, and D correspond to the semester during which each course should be taken. For example, Computational and Mathematical Thinking 2 (CMT 2) will be offered during Semester B and C (the second and third semester of the program sequence).

4 A New Approach to K-12 Computer Science Education Teacher Preparation

At the foundation of the proposed program, and the nine courses that comprise the core of the program, is a comprehensive approach to teacher preparation in computer science education (see Figure 1).

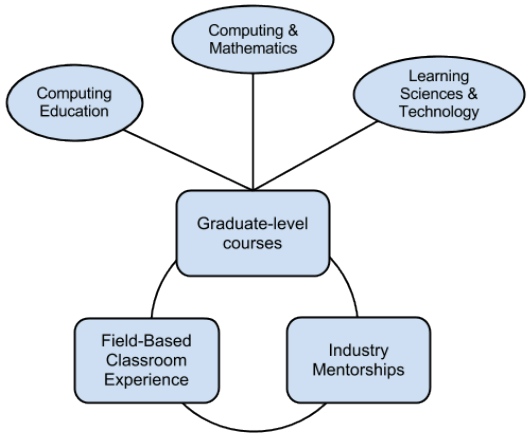

Figure 1: Computing Education Degree: Conceptual Framework

4.1 Graduate-Level Courses

The proposed courses for the Computing Education degree consists of courses across three sub-groups: Computing Education, Computing & Mathematics, and Learning Sciences & Technology.

4.1.1 Computing Education (CE)

In the *Computing Education (CE)* sub-group the two proposed courses, Pedagogical Practice & Methods in Computation & Mathematics (PMCM) and Practicum in Computing Education (PCE), provide both in-service and pre-service teachers with the experience and theoretical foundation to develop independent instructional lessons for

primary and secondary school students using grade-level appropriate content and validated 'best practices' to inform pedagogical classroom design.

4.1.2 Computing & Mathematics (C&M)

The *Computing & Mathematics (C&M)* sub-group contains a series of courses, Computational & Mathematical Thinking (CMT) 1, 2, and 3, that cover a wide range of core computer programming, computer science, software engineering, and mathematical thinking topics. As the core content courses for the degree, CMT1, 2, and 3 are designed to establish a solid foundation in computing while simultaneously introducing key mathematical thinking principles to ensure teachers are prepared to teach abstractions in the K-12 classroom. In addition to these core content courses, the C&M subgroup contains courses that cover discrete structures, algorithms, and computational science with a focus on 'big data.'

4.1.3 Learning Sciences & Technology (LS&T)

The *Learning Sciences & Technology (LS&T)* sub-group contains one course, Foundations: Learning Sciences & Technology, which provides all students with a solid foundation in the central disciplines (computer science, educational psychology, cognitive psychology, and applied linguistics) that define the learning sciences within the context of computer science education.

4.2 Field-Based Classroom Experience

In addition to the proposed sequence of courses, each student will be placed in a semester-long classroom-based teaching experience (in the case of pre-service teachers) or will work alongside a faculty mentor to augment a current CS education classroom experience (for in-service teachers). The experience, part of the CE sub-group course 'Practicum in Computing Education (PCE),' ensures that all teachers are receiving authentic teaching and learning experiences as they develop their content and pedagogical knowledge expertise.

4.3 Industry Mentorships

The third major component of the Computing Education Degree: Conceptual Framework is the placement of a teacher in an industry mentorship. As a degree of study that is connected to professional industry, the Industry Mentorship ensures that teachers are creating authentic learning experiences for the students while simultaneously gaining professional experiences themselves. In states where industry experience is critical to teacher certification in the technology fields, such as New York State, Industry Mentorships help to establish the professional foundation upon which a teacher can effectively build academic experiences for their K-12 classroom.

5 Conclusion

Developing a comprehensive master's degree for in-service and pre-service teachers in K-12 computer science education ensures that teachers are being prepared to enter the classroom ready to offer effective instruction. As a conceptual model, the proposed program in Computing Education provides colleges and universities with a comprehensive outline for designing and developing a rigorous sequence of graduate-level instruction that ensures teachers, upon completion of the course, will possess the requisite knowledge needed to teach computer science and software engineering at the K-12 level.

6 References

[1] Bureau of Labor Statistics, U.S. Department of Labor, Occupational Outlook Handbook, 2012-13 Edition, Software Developers, on the Internet at http://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm.

[2] Margolis, J., Estrella, R., Goode, J., Jellison-Holme, J., & Nao, K. (2008). *Stuck in the Shallow End: Education, Race, & Computing*. MIT Press: Cambridge, MA.

[3] Gal-Ezer, J. & Stephenson, C. (2010). Computer Science Teacher Preparation is Critical. ACM Inroads, Vol. 1(1), 2010, 61-66.

[4] Purdue University Computer Science Teaching Endorsement (2010) Accessed from http://www.cs.purdue.edu/news/11-05-10CS4EDU.html

[5] Nova Southeastern University (nd) Education Specialist Program in Computer Science Education, Accessed from http://www.fischlerschool.nova.edu/es/computer/degreeonly

[6] Purdue University (nd) *Computer Science Teaching Supplemental Licensure Program*, Accessed from http://cs4edu.cs.purdue.edu/license

[7] Guzdial, Mark (2011) *Whats the argument for becoming a computer science teacher?*, Accessed from http://computinged.wordpress.com/2011/02/07/whats-the-argument-for-becoming-a-computer-science-teacher/

[8] Wilson, C., Sudol, L., Stephenson, C., & Stehlik, M. (2010) *Running the Empty: Failure to Teach K-12 Computer Science in the Digital Age*. Association for Computing Machinery

[9] Khoury, G. (2007). Computer Science state certification requirements. CSTA Certification Committee Report. Retrieved June 11, 2009, from http://www.csta.acm.org/ComputerScienceTeacherCertification/sub/TeachCertRept07New.pdf

[10] National Academies Press (2000). *How People Learn: Brain, Mind, Experience, and School* National Research Council

[11] Center for Urban Science and Progress (2012) *Mayoral Press Conference Announcing Launch of CUSP*, Accessed from http://cusp.nyu.edu/mayoral-press-conference-announcing-launch-of-cusp/

[12] Kaminer, April (2013) *New Cornell Technology School Tightly Bound to Business*, Accessed from http://www.nytimes.com/2013/01/22/nyregion/cornell-nyc-tech-will-foster-commerce-amid-education.html

[13] Herman, C., Tucker, D., (2012) *Teaching a New Generation of Coders and Web Developers*, Accesed from http://www.wnyc.org/shows/newtechcity/2012/sep/11/

[14] New York City Department of Education (2013) *Mayor Bloomberg and Schools Chancellor Walcott Announce the 20 Schools Selected for the New Software Engineering Pilot Program*, Accessed from: http://schools.nyc.gov/Offices/mediarelations/NewsandSpeeches/2012-2013/AppliedSciences.htm

[15] Miller, C.C. (2013) Opening a Gateway for Girls to Enter the Computer Field. The New York Times, Accessed from http://dealbook.nytimes.com/2013/04/02/opening-a-gateway-for-girls-to-enter-the-computer-field/?src=twrhp

[16] Guzdial, M. (2012) Biggest challenge for CS10K: Recruiting the Teachers. Access from http://computinged.wordpress.com/2012/02/09/biggest-challenge-for-cs10k-recruiting-the-teachers/

[17] Guzdial, M., Bruckman, A., Ericson, B., & McKlin, T. (2011) Project "Georgia Computes!" An NSF Broadening Participation in Computing Pipeline Alliance, Retrieved from http://gacomputes.cc.gatech.edu/Members/guzdial/GaComputes-2pgr-Jan11-CE21Mtg-smaller.pdf

[18] Cuny, J. (2011) Transforming Computer Science Education in High Schools. IEEE Computer Society, p. 107 - 109.

[19] Depillis, L. (2013) How to Close the Tech Industry's Gender Gap. The New Republic, Retrieved from http://www.newrepublic.com/article/112784/tech-industry-gender-gap-closing-it-starts-classroom#

[20] Fadjo, C. L. (2012) Developing Computational Thinking Through Grounded Embodied Cognition. Unpublished Doctoral Dissertation. Columbia University.

[21] Grover, S. & Pea, R. (2013) Computational Thinking in K-12: A Review of the State of the Field. Educational Researcher, Vol. 42 No. 1, pp. 38 - 43.