Workshop Proposal
Computational Thinking for Practicing Engineers

Orieta Celiku
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
+1 412 268 3501
orietac@cs.cmu.edu

David Garlan
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
+1 412 268 5056
garlan@cs.cmu.edu

Anthony Lattanze
Institute for Software Research
Carnegie Mellon University
Pittsburgh, PA 15213
+1 412 268 4736
lattanze@cs.cmu.edu

SUMMARY
This workshop will address the problem of providing education in foundational computer science principles and techniques for the large population of practicing software developers who do not have a formal computer science background. Participants will be introduced to an NSF-sponsored project within Carnegie Mellon University's Masters in Software Engineering Programs to bridge gaps in formal training of practicing engineers using modules that cover the foundational material in a way that can be put to use directly. A key feature is tight coupling with practical exercises within the context of hands-on applications. Participants are encouraged to contribute ideas regarding content, delivery methods, outcomes, and examples.

1. OVERALL OBJECTIVES
Advances in the software industry have led to an increasing need for *computer professionals* with solid foundational knowledge in computational thinking [2]. Unfortunately many professionals do not have this training, having entered the field of software development through other disciplines.

Currently suitable educational alternatives do not exist for this population. Taking a large number of undergraduate courses in foundational computer science topics is inappropriate because of the time commitment and the fact that such courses are oriented towards students with little experience or domain expertise. Moreover, most graduate courses typically assume this knowledge as a prerequisite.

We propose to address this critical need by condensing the foundational knowledge into a set of modules that streamlines its presentation and relates it to practical engineering examples, targeting students with some experience in software development. If successful, such a course will help bridge the gap between the practical nature of today's software systems and the foundations needed to apply advanced techniques and theories to solve real problems.

This workshop will explore the problem of providing educational alternatives that would allow practicing engineers to acquire computational thinking fundamentals. Through guided discussion, the workshop will consider the problem and possible solutions, informed by our initial experience in developing an NSF-sponsored bridge curriculum as part of the Masters in Software Engineering Programs at Carnegie Mellon University [1].

2. OUTLINE OF THE WORKSHOP
The workshop will be organized in three main parts:

(1) Presentation of the problem and requirements for solution

(2) Description of the work we have done so far to develop a new course addressing those requirements

(3) Discussion of important issues for this class of education

2.1 Rough Agenda
1. Introduction to the problem

   In this part of the workshop we will outline the need for computational thinking skills for modern computer professionals, particularly in addressing modern engineering concerns, such as scalability, performance, and reliability. We will also address problems with existing educational alternatives in meeting this need.

2. Ongoing efforts to bridge the gaps in the formal training of practitioners

   In this part of the workshop we will present CMU's initial efforts to develop a bridge course that addresses the gap in educational alternatives for practitioners. Key features of the approach include (a) distilling and packaging the set of core, foundational concepts of computational thinking that have stood the test of time and are relevant broadly to professionals working in software development or other disciplines that have a significant computational component; (b) grounding the material in practical applications so that students can see how these concepts apply in practice; and (c) streamlining of the course content to take advantage of the professional maturity of the students.

3. Open discussion

   In this part of the workshop we will lead a discussion addressing questions such as
   - What areas of computer science are most essential and/or poorly understood by practitioners?
   - What competency level should the course aspire to?

- How should the presentation of course materials be streamlined to take advantage of the students' experience?
- What kinds of real world examples are good vehicles for conveying the foundational concepts?

4. Wrap-up, conclusions, and next steps

3. EXPECTATIONS

During the workshop participants will be introduced to this area and exposed to the materials developed at Carnegie Mellon, including course syllabus, readings, and exercises grounded in practical systems.

A workshop report summarizing the discussion will be distributed to the participants at a later date.

We expect the workshop to generate interest in further discussions and, possibly, lead to additional meetings and publications.

We expect the discussion to clarify the goals and techniques for courses similar to the one we have developed.

We hope that this workshop will generate interest in addressing this educational problem and, in the fullness of time, lead to new educational initiatives at other universities and institutions.

4. LOGISTICS

This will be a 3-hour workshop, with a target attendance of 20 participants. The NSF grants sponsoring this project has provided funds to support partial travel and expenses for some participants. We plan on making specific invitations to key participants, who we hope to attract to the workshop and the conference as a whole.

5. SUITABILITY FOR A SPECIAL WORKSHOP AT CSEET

This conference is a natural venue for discussing how to meet the computer science educational needs of practicing engineers. We expect many experienced software professionals to attend the conference and have useful input.

Within the context of CSEET a workshop seems an ideal venue for this activity because it allows for both presentations of existing material and discussion of ideas that could lead to enhancements and new approaches.

ACKNOWLEDGMENTS

The development of the Computational Thinking for Practicing Engineers course is supported by NSF under grant CNS0836133.

6. REFERENCES

[1] D. Garlan, D. Gluch and J. Tomayko. "Agents of Change: Educating Software Engineering Leaders." In *IEEE Computer*, Vol. 30/No. 11, pages 59-65, 1997.

[2] J. Wing. "Computational Thinking." *Communications of the ACM*. Vol. 49/No. 3, pages 33-35, March 2006.