

Colloquium

Department of Computer Science

Dr. Venera Arnaoudova

Venera Arnaoudova is a postdoctoral research fellow at the University of Texas at Dallas, working with Dr. Andrian Marcus. She received her Ph.D. degree in 2014 from Polytechnique Montréal under the supervision of Dr. Giuliano Antoniol and Dr. Yann-Gaël Guéhéneuc. Her research interest is in the domain of software evolution and particularly, the analysis of source code lexicon and documentation. Her dissertation focused on the improvement of the code lexicon and its consistency using natural language processing, fault prediction models, and empirical studies. Arnaoudova has published in several international software engineering conferences and journals, earning a Best ERA Paper Award at the 26th IEEE Conference on Software Maintenance (ICSM 2010). She has been part of the organizing committee for the Working Conference on Reverse Engineering (WCRE 2013), the International Conference on Software Testing, Verification and Validation (ICST 2012), and several workshops and summer schools. She also served as program committee member for the International Conference on Program Comprehension (ICPC 2015); the Early Research Achievements Track at the International Conference on Software Analysis, Evolution, and Reengineering (SANER 2015), and ICPC 2013. More information available at: <http://www.veneraarnaoudova.com>

March 27, 2015

2:00 p.m.

Guyon Auditorium, Morris Library

Lexicon Quality in Software Evolution

Abstract

Program comprehension is a key activity during software evolution and maintenance. It is an extremely challenging activity and frequently performed by developers—more often than code writing. The more complex the software is, the harder it is to comprehend it and the higher the chances are to induce errors while changing it. Hence, structural properties such as size and complexity are often used to identify complex and fault prone programs. Existing studies of developers' behavior while understanding a program show that the textual information contained in identifiers and comments—i.e., the source code lexicon—is one of the factors affecting the psychological complexity of software. Psychological complexity, like structural complexity, makes a program difficult to understand and to maintain by developers. In this talk, we discuss how to assess the psychological complexity of a program using lexicon-based metrics. We also discuss how the quality of the source code lexicon relates to the quality of the software and program comprehension. Finally, we present techniques and tools that leverage the source code lexicon to support developers in better performing important software engineering tasks, such as, re-documentation and code reviews.