Title: Probabilistic Symbolic Execution

Speaker: Matthew Dwyer  
Henson Professor and Acting Chair  
Department of Computer Science and Engineering  
University of Nebraska - Lincoln  
http://www.cse.unl.edu/~dwyer

Date/Time: 26 June 2014, Thursday, 03:30 PM to 05:00 PM

Venue: Executive Classroom, COM2-04-02

Chaired by: Dr Rosenblum, David S., Professor, School of Computing  
(david@comp.nus.edu.sg)

Abstract:

Symbolic execution has been applied over the last decade to automate the generation of test suites, perform precise differencing of software versions, and detect faults and security vulnerabilities. Symbolic execution is path-sensitive, but it treats all paths equally, when developers understand that some paths will be executed much more frequently either because specific inputs are more likely or the sets of values that drive execution down a path are large. In this talk, we describe recent work that incorporates probabilistic reasoning into symbolic execution to quantify the likelihood of executing a program path. We describe how this is done for sequential programs in some detail and sketch more recent work for non-deterministic and concurrent programs. Applications of these techniques include being able to quantify the confidence that a property holds when it cannot be proven definitively, ranking errors based on their likelihood, and accelerating statistical program analysis.

Biodata:

Matthew B. Dwyer is the Leonard A. Lovell Professor of Engineering and Acting Chair of the Department of Computer Science and Engineering at the University of Nebraska-Lincoln. He received a Bachelors in Electrical Engineering from the University of Rochester (1985), a Masters in Computer Science from the University of Massachusetts at Boston (1989), and a Doctorate in Computer Science from the University of Massachusetts at Amherst in 1995. From 1985 through 1990 he worked at Intermetrics Inc. on the development of embedded systems and compilers. He was a faculty member at Kansas State University and currently holds an appointment as Extraordinary Professor at Stellenbosch University, South Africa.
He has published widely in the areas of program analysis, software specification, and automated formal methods. His work has received several awards including the ICSE "Most Influential Paper" and SIGSOFT "Impact Paper" awards in 2010 and he has been named ACM Distinguished Scientist (2007), Fulbright Research Scholar (2011), and IEEE Fellow (2013). Dr. Dwyer has served as program chair of a number of top research conferences including: the ACM SIGSOFT Symposium on Foundations of Software Engineering (FSE 2004), the International Conference on Software Engineering (ICSE 2008), and the ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA 2012). His editorial work includes serving as co-Editor-in-Chief for the International Journal of Software Tools for Technology Transfer (2001-2007), Associate Editor of ACM Transactions on Programming Languages and Systems (since 2009), Editorial Board Member for CACM Research Highlights (since 2013), and Associate Editor (2006-2009) and Editor-in-Chief (since 2014) of IEEE Transactions on Software Engineering.