NATIONAL UNIVERSITY OF SINGAPORE

School of Computing

C S   S E M I N A R

Title: **From Calculus to Computation**

Speaker: Associate Professor Olivier Danvy
              Arhus University, Denmark

Date/Time: 6 November 2015, Friday, 02:00 PM to 03:30 PM

Venue: Executive Classroom, COM2-04-02

Chaired by: Dr Khoo Siau Cheng, Associate Professor, School of Computing
              (khoosc@comp.nus.edu.sg)

Abstract:

On one hand, there is the notion of calculus: terms and algebraic operations, that together with a strategy to apply these operations step by step, define a notion of computation. On the other hand, there are computers that carry out computations in big steps. And joining these two hands, there are programming languages to express the computations that should be carried out.

It's a nice picture. For example, one likes to say that the lambda-calculus (which is a calculus with terms, operations, and strategies) is the foundation of functional languages. By analogy, one can develop a calculus of objects as a foundation for object-oriented languages. However, when implementing a programming language, one may wonder where the calculus has gone. For example, for functional languages, the lambda-calculus's beta-reduction is only used as a metaphor since lambda-terms are compiled away, and conversely, what is the calculus counterpart of the all-important garbage collector at run time? Likewise, stack inspection is formalized with a secure lambda-calculus in theory, but in practice it is implemented with security-passing style. How do we know that one corresponds to the other?

The goal of this presentation is to show how calculus morphs into computation. Starting from a representation of calculus based on reduction ("reduction-based evaluation"), one can methodically liberate oneself from this small-step representation to end up with a big-step representation of computation based on evaluation ("reduction-free evaluation"). The presentation concludes on the effectiveness of this morphing with an array of examples.

Biodata:

Olivier Danvy is an Associate Professor at Arhus University, and currently a Visiting Associate Professor in the Department of Computer Science for the fall of 2015-2016. He is teaching CS6202 (Advanced Topics in Programming Languages) and is also conducting an uncredited, but well-attended (considering) seminar on Communication in Computer Science (danvy.org/tips-and-tricks). He is interested in all aspects of programming languages, from their logic and semantics to their implementation, including programming.

Olivier has contributed to functional programming, partial evaluation, continuations, and program transformations, and for over 10 years, he has co-edited the Kluwer/Springer journal "Higher-Order and Symbolic Computation". His h-index is over 30, and he is one of those rare computer scientists who has more journal publications than conference publications. Ten years ago, he was measured to be the most acknowledged scientist in the CiteSeer data base (Giles and Councill, "Who gets acknowledged: Measuring scientific contributions through automatic acknowledgment indexing", Proceedings of the National Academy of Sciences of the United States of America 101(51):17599-17604, December 21, 2004 and Nature, 432:790, December 16, 2004). He has supervised over 20 PhD students, totaling over 12 distinct nationalities.