Queen's SCHOOL OF Computing

# Seminar

**Dr. Joshua Dunfield**
Researcher
University of British Columbia

**Friday, May 19 2017**
**2:30-3:30 pm**
**Dupuis Hall 215**
*Light Refreshments*

## Gradual typing for refinements

Writing good software is hard. Much of the thought and expertise that goes into designing a program isn't recorded formally. My research gives programmers the means to express their intent, and the tools that use that intent to improve software.

In this talk, I'll focus on making datasort refinements easier to use. Refinements express more precise invariants than ordinary static types do, but this precision makes them harder to apply in practice. Gradual typing was originally developed to allow smooth migration from a dynamically typed program to a statically typed program, but the underlying idea also applies to adding refinements to a statically
typed program. I will present recent work on gradual typing for datasort refinements, and discuss ongoing work that I hope will serve as a unified foundation for a variety of gradual type systems.

**Joshua Dunfield** is a researcher at the University of British Columbia. He works on programming languages from a type-systems perspective, from refinement types (more precise static typing) to type-based incrementalization (using type information to generate an incremental program). After receiving a PhD from Carnegie Mellon for work on refinement typing, he went to McGill to work on dependently typed programming languages, then to MPI-SWS in Kaiserslautern to work on incremental computation. Having undergone three international moves in the last decade, he is looking forward to staying in Canada for the foreseeable future.