CSci 5980 and 8980:
Specifying and Reasoning about Software Systems
Spring 2017
Instructor: Gopalan Nadathur
http://www.cs.umn.edu/~gopalan

Course Overview: The course will introduce a logical approach to specifying, prototyping and reasoning about formal systems that are typically described via syntax-directed rules. This style of presentation has been used, for example, for typing and evaluation relations for programming languages, for proof systems for varied logics, for software modelling languages and software specifications, and for process calculi and encodings of concurrent systems. We will initially expose a specification language based on a higher-order logic that provides a natural and effective means for formalizing such systems. We will then consider logical devices and systems for reasoning about formalizations in the specification language. Finally, we will understand the two-level logic approach that smoothly integrates specification, prototyping and reasoning and also allows meta-theoretic properties of the specification logic to be used to advantage in reasoning. The Teyjus, Bedwyr and Abella systems that implement the specification and reasoning logics will be used to provide concreteness to the foundational and methodological discussions. We also expect to explore the relationship of the main methods to be studied to methodologies based on dependently typed lambda calculi that are used in systems like Twelf and Coq.

Course Format and Required Work: The course will be organized in a pseudo-seminar style. The instructor will deliver lectures on the basic material. There will be regular (mandatory) reading assignments and a few written assignments. Each student taking the CSci 5980 version will additionally be required to complete an individualized programming or theorem-proving project on a topic related to the material covered in class. For the CSci 8980 version, students will be required to carry out a more advanced, independent study with some original content and to prepare a term paper that they will present in class. Grades will be determined by attendance, which will be required, satisfactory fulfillment of the reading and written assignments, and the quality of work manifest in the term paper/project and the class presentation as relevant.

Sources for the Course: The course will be based on material in research papers and in the book entitled Programming with Higher-Order Logic co-authored by the instructor.

Prerequisites: The main requirements are mathematical maturity, a prior exposure to logic and the capability to program in an advanced language like OCaml, Standard ML or Haskell. If you are not sure that you meet these requirements, talk to the instructor to determine if you would still be able to feel comfortable in the course.

Time and Place: MW 9:45-11:00, 2-260 KH Keller Hall.