Have you ever needed to prove a property about a formal system but did not know where to start or what techniques to use? Have you ever had trouble sleeping, worrying whether you made a mistake in a paper you just submitted to a conference? If so, then this course is for you. The course is a hands-on introduction to theorem proving using the Isabelle proof assistant. The course will cover reasoning about inductive datatypes, recursive functions, and sets using higher-order logic. The course will cover common proof techniques including a cornucopia of induction methods. Last but not least, this course will cover practical issues such as how to “debug” your proofs, techniques for structuring large proofs, how to make the best use of Isabelle’s decision procedures, and how to get Isabelle to typeset your definitions and proofs in LaTeX. The course has no explicit prerequisites, but a basic understanding of logic is helpful, as is some exposure to functional programming.

**When** 1-1:50 MWF

**Where** ECEE 265

**Syllabus**

1. Programming in Isabelle.
2. Writing proofs in Isabelle.
3. Proof by induction.
4. More proof techniques (proof by contradiction, etc.).
5. Advanced forms of induction.
6. Case study: compiling to a stack machine.
7. Reasoning about sets.
9. Inductively defined sets and graph algorithms.
10. Case study: the simply typed lambda calculus.