

<b>Course Title:</b>	<b>CS-840, Software Verification</b>
<b>Credit Hours:</b>	3+0
<b>Pre-requisites:</b>	<ul style="list-style-type: none"> <li>▪ Software Engineering</li> </ul>
<b>Course Description:</b>	This course provides details about the formal techniques for proving programs correct, checking consistency and completeness. The course is intended to bring students into contact with existing verification techniques and current research topics in the field of theorem proving and automated deduction and to teach them the necessary skills to successfully use industrial grade verification environments in modeling and verification.
<b>Tools and Technologies:</b>	<ul style="list-style-type: none"> <li>▪ Based on instructor preference</li> </ul>
<b>Learning Outcomes:</b>	<p>On successful completion of this course students will be able to:</p> <ol style="list-style-type: none"> <li>1. Realize the importance of formal methods in developing safety critical applications.</li> <li>2. Acquire in depth knowledge of concepts related to safety and reliability analyses of software</li> <li>3. Apply proof techniques in establishing correctness of formal software requirements specifications.</li> <li>4. Prepare software requirements specifications using an industry standard.</li> <li>5. Understand and identify current research challenges in the area.</li> </ol>
<b>Tentative MS Thesis:</b>	<ul style="list-style-type: none"> <li>▪ Semantic Infrastructure for Model-based Integration</li> </ul>
<b>Text Books:</b>	<ul style="list-style-type: none"> <li>▪ Clarke, Peled, Grumberg (1999): Model Checking, , MIT Press</li> </ul>
<b>Reference Books:</b>	<ul style="list-style-type: none"> <li>▪ Baier, Katoen (2008): Principles of Model Checking, , MIT Press</li> <li>▪ Berard, Bidoit, Finkel, et al (2001): Systems and Software Verification: Model-Checking Techniques and Tools. Springer.</li> </ul>
<b>Course Contents:</b>	<ul style="list-style-type: none"> <li>▪ Background to Logic as a Software Specification Language</li> <li>▪ White-box/Black-box Testing based on Code Analysis</li> <li>▪ Models and Semantics</li> <li>▪ Temporal Logic based Verification</li> <li>▪ Typed Lambda Calculus, Higher-Order Logic</li> <li>▪ Natural Deduction in HOL</li> <li>▪ Term Rewriting</li> <li>▪ Advanced Topics</li> </ul>