

Software Verification

Syllabus Information

CS 3450 - Software Verification

Associated Term: 2023/24 Academic Session

Learning Objectives:

We study the formal analysis of computer programs using types, logic, and automata based-approaches. Students will learn to relate computer programs to different models of program behaviour such as process calculi and transition systems. The role of abstraction will be explored. We will discover how requirements can be expressed using formal techniques. These include automata, logic, and types based approaches. These approaches are often closely related and examples of the translations between the representations will be described. We will learn manual and automatic techniques to verify a program model against its specification. The module will be accompanied by laboratory sessions which will use appropriate software tools to explore the concepts covered by the module. Pre-requisites: CS1860, CS1870 Learning Outcomes: 1. Appreciate the need for verifying the correctness of systems against formal specification of requirements 2. Express requirements on programs formally using logic, automata, and type-based approaches. 3. Understand different models of computer programs. 4. Use abstraction to formalise the behaviours of programs. 5. Learn how to reason whether a program model matches its specification. 6. Gain practical experience of formal analysis using appropriate tools.

Required Materials: [Click here for the reading list system](#)

Technical Requirements: The total number of notional learning hours associated with course are 150. These will normally be broken down as follows: 22 hour(s) of Lecture(s) across 11 week(s) 22 hour(s) of Laboratory across 11 week(s) 102 hours of Guided Independent Study Formative Assessment: Laboratory classes - Oral feedback on exercise sheets Summative Assessment: Coursework 20% Examination 80% 2 hours