Leveraging Association Between Complexity and Bugs for Testing
Tung Nguyen
Mentor: Professor Iftekhar Ahmed.

Software will always have bugs, and as software continues to become a more and more pervasive part of our lives, software bugs will continue to affect more people than ever. As software systems get more complex, the task of exhaustive testing becomes more complex and even infeasible in some cases. In order to build less error-prone systems, we, therefore, need to focus on quickly and efficiently testing error-prone parts of the code. Our research investigates using software complexity as a guiding mechanism for focusing testing efforts. Prior research has investigated using cyclomatic complexity for this purpose since cyclomatic complexity is associated with bugs. However, recent advances in software complexity measurements have identified new software complexity metrics such as npath complexity, and path complexity which are better measurements of software complexity. Association of these newly identified metrics with software faults is yet to be investigated. In this research, we analyze the association between newly identified software complexity metrics (i.e., npath complexity, path complexity) and known faults. We also analyze the association between cyclomatic complexity and known faults as a baseline for comparison. We are using a well-known database of faults named Defects4j as our benchmark which consists of 483 known faults. After conducting measurement for 65 bugs, we have found a high correlation between newly identified complexity metrics and bugs. Further data is being collected where we analyze all 483 bugs and the results will be presented at the symposium. If the high correlation between bugs and metrics persists for all bugs, then the next step would be to use these metrics for test case generation and prioritization.