PREFACE

Program Analysis is the process of automatically analyzing the behavior of computer programs. Static and Dynamic analysis are two types of program analysis. Analyzing computer software without actually executing programs is static analysis. The analysis of computer software that is performed while executing programs is dynamic analysis. Program invariants play a very important role in almost all the phases and aspects of software engineering. Program invariants are very much useful in software development. They protect programmers from making errant changes and verify properties of a program. Invariants can be explicitly stated in programs. Programmers can annotate code with invariants but this takes time and effort and also many significant invariants will be overlooked. Invariants can be detected using two types of techniques, static and dynamic. But, dynamic methods are more effective when compared to static methods. Diduce, carrot, arnout’s, henkel and diwan, daikon are some of the tools through which invariants can be detected dynamically. Among these tools daikon is the most efficient and extensively used tool. However, it has drawbacks, which affects its speed and performance.

Software Evolution is the process of developing software initially, then repeatedly updating the software. There will be many reasons which lead to software evolution and some of them will be various types of software maintenance. During software evolution invariants help a lot in decision making but analyzing all invariants which include even unnecessary invariants is waste of time and effort; also they may lead to wrong decisions. So, to support software evolution it is required to reduce the number of invariants. Further, in the list of relevant invariants it is not required to analyze all invariants. It is sufficient to consider and analyze the invariants which are being affected by the change. So, with this intension to support software evolution by reducing time, effort and hence cost, a methodology is utmost necessary. This has been considered and addressed in a successful manner in this thesis as can be found all through the chapters.