Motivation and Background
As the complexity of software increases, designing and developing new software systems becomes more challenging. To handle this complexity, there is a trend to partially automate software development tasks supported by optimization methods. This area is known as Search Based Software Engineering (SBSE) [Har07a].

Up to 90% of the software expenses [LGFW13] are invested to fix software defects. This activity consists of many subtasks including fault localization, patch generation and implementation, and execution of regression test cases. Mechanizing these tasks has received significant attention from the automated and search based software engineering community. Approaches developed by Le Goues et al. [WNLGF09, LGNFW12, LGDVFW12], and Martinez and Monperrus [DMM’15, MM15, XMD’17] and Kim et al. [KNSK13] show meaningful results towards the automation of bug fixing. The main idea of these approaches is to use failed test cases to localize potential fault and then apply mutations to the source code until the program satisfies all unit test cases. The mutation that are applied to the program code can range from small changes like modification, addition or removing a single code lines [WNLGF09, LGNFW12] to complex edit operations [MM15, KNSK13] that have been mined from software repositories and that have been used to fix a fault in a different context. However, there are still many challenges that need to be resolved in order to reach higher scalability, efficiency and quality.

Goals
The goal of this project is to analyses the search space for common automated program repair problems.

Description of the Task
- Understand the current problems in automated program repair
- Run and analyses experiments in the area of automated program repair
- Provide information/characterization about the search spaces in this area

Research Type
Theoretical Aspects: ****
Industrial Relevance: ****
Implementation: ****

Prerequisite
The student should be enrolled in the bachelor/master of computer science program, and has completed the required course modules to start a bachelor/master thesis.

Skills required
Programming skills in Java or C++, understanding of, or willingness to learn, the software engineering and software analysis foundations needed for the project.

Contacts
Lars Grunske, Humboldt-Universität zu Berlin, Institut für Informatik, Lehrstuhl Software Engineering, Unter den Linden 6, 10099 Berlin, Germany

Application
Please contact me during my office hours or send me an email with the title: “[ThesisProject]-SC4SBSE-AutoRepair” to se-career@informatik.hu-berlin.de
References


