

---

**Abstract**

Unit testing has a significant role in software development and its impacts depend on the quality of test cases and test data used. To reduce time and effort, unit test generator systems can help automatically generate test cases and test data. However, there is currently no unit test generator for Kotlin programming language even though this language is popularly used for android application developments. In this study, we propose and develop a test generator system that utilizes genetic algorithm (GA) and ANTLR4 parser. GA is used to obtain the most optimal test cases and data for a given Kotlin code. ANTLR4 parser is used to optimize the mutation process in GA so that the mutation process is not totally random. Our model results showed that the average value of code coverage in generated unit tests against instruction coverage is 95.64%, with branch coverage of 76.19% and line coverage of 96.87%. In addition, only two out of eight generated classes produced duplicate test cases with a maximum of one duplication in each class. Therefore, it can be concluded that our optimization with GA on the unit test generator is able to produce unit tests with high code coverage and low duplication.

**Keywords:** unit test generator, test case, optimization, Kotlin, genetic algorithm (GA)

---