Code Duplication Checker

Jennifer Galley - Abebe Biru - Rithesh Baradi - Rishi Barua - Sachin Sebastian

# Motivation

<why is this project worth doing?>

We wanted to experiment with detecting duplicate code in C# because we were inspired by the amount of duplicate code we encountered in our day-to-day work, and the automation possibilities for detecting such clones. There are many tools available which implement clone detection on C#, but we thought it would be interesting to start from scratch and compare different implementations and algorithms. We also discovered an efficient, language-agnostic implementation in the research, initially implemented in Java, called Count Matrix Clone Detection which we thought would prove interesting to recreate in C#.

# Design Decisions

<why did you implement the prototype the way you did?>

We made the project in C#, using the latest version of .NET, as a console application. This was because we are all familiar with C#, Visual Studio, and running console applications. We decided to implement the different similarity algorithms (AST, CMCD) as different projects so that they could be modular and interchangeable. The main project is responsible for parsing the inputs, calling the similarity algorithm, and visualizing the results. For storing and displaying the results, we chose to use a class which had the filename, start line, end line, and block of duplicate code so that we could try to mimic popular code editor’s visualizations for git diffs when building the HTML display of the duplicates.

# Testing and Debugging

<what testing/debugging approaches did you use? What unique challenges did you face?>

We used unit tests to test our project. Some of the harder challenges we faced were building fake duplicate code for testing and being able to see the differences between the expected and actual results in a huge block of HTML.

# Evaluation

<how did you evaluate whether your prototype improves over existing tools?>