

Using contracts such as method preconditions, postconditions, and class invariants in code is an approach to preventing bugs in object-oriented programs. In the .NET Framework, this is possible due to the framework of Code Contracts, which includes Clousot, a tool for static program analysis based on abstract interpretation. Although string is a fundamental type in .NET programs, Clousot does not have a usable support for analysis of string values. In this thesis, we explore the specifics of string manipulation in the C# language and in the .NET Framework, and show how they can be covered by static analysis. Our approach is to use the methods of the `String` class and a subset of regular expressions to specify string properties in code, and to use abstract interpretation with non-relational abstract domains to reason about those properties. We chose a small number of already published abstract domains for strings, which vary in their complexity and ability to represent different properties. We adapted those domains to our setting, including definitions of abstract semantics for the supported string methods. We implemented the abstract domains in Clousot in a way that cooperates with numerical analysis and allows adding more string abstract domains in the future.