# Installation

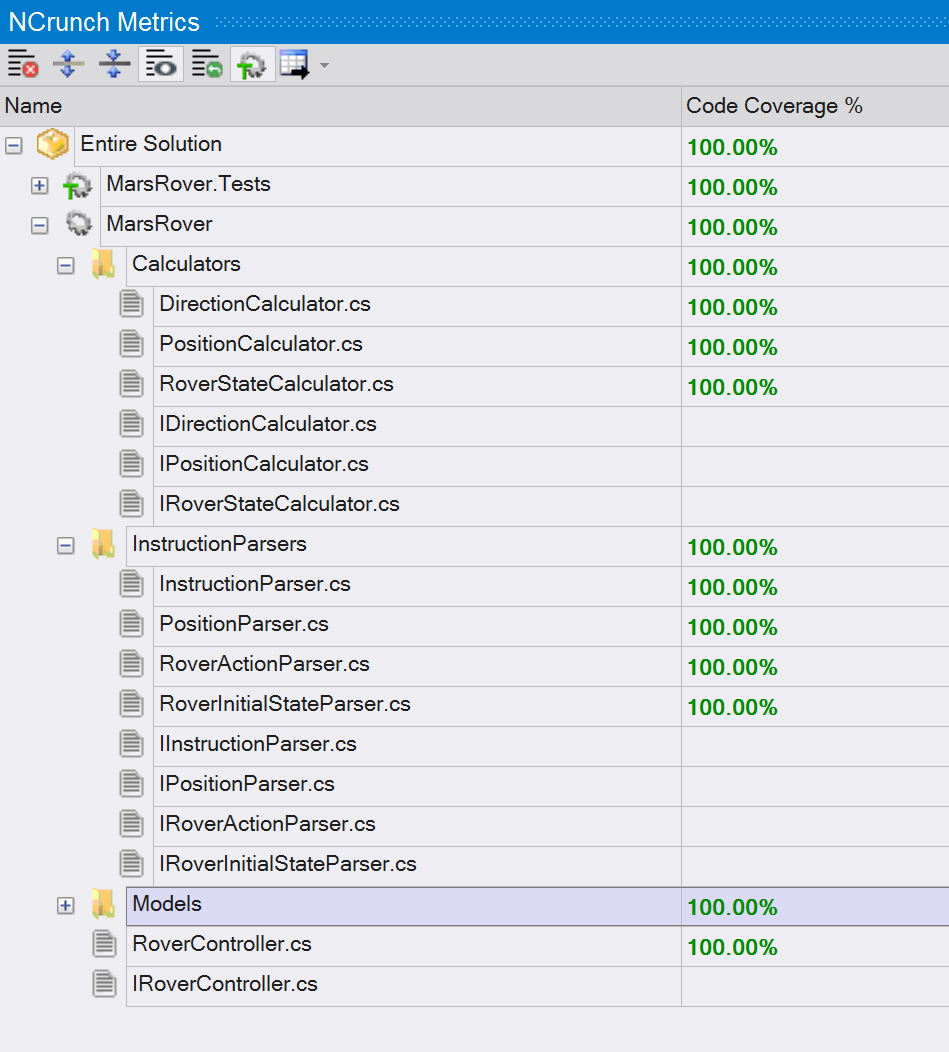
Please make sure .NET Core 3.0 SDK or above is installed.

# Libraries used

NSubstitude and xUnit are used for unit tests.

# Unit test coverage

This project has 100% unit test code coverage

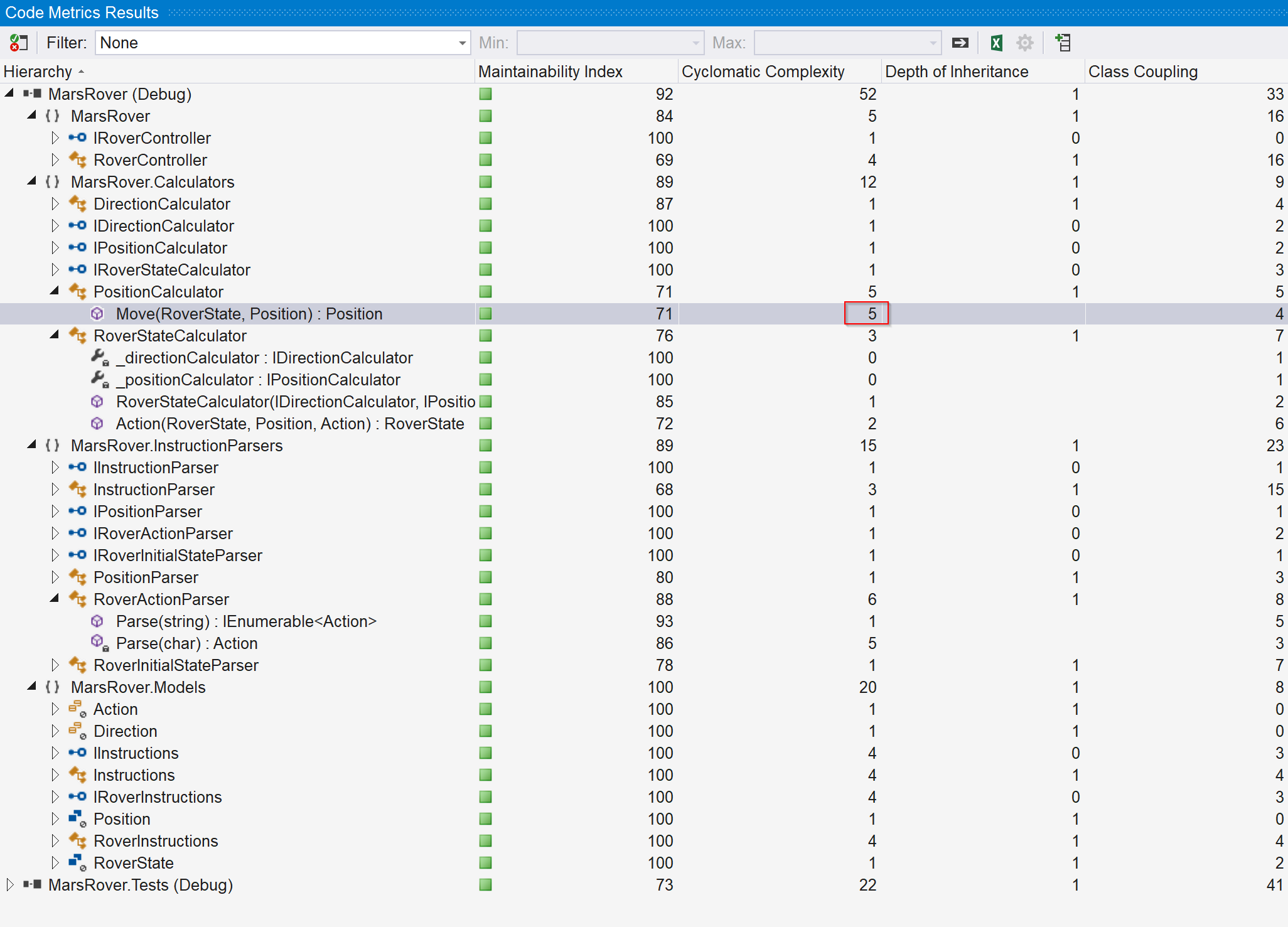


## Design Considerations

I used lots of interfaces. This is to comply Dependency Inversion principle, Open-Closed Princple. Small interfaces are used to comply with Interface Segregation Principle. Functions and classes are small because of Single-Responsibility Principle.

My functions are small, focused, easy to understand. Functions and classes are decoupled.

Maximum cyclomatic complexity of all functions is 5:



No inheritance is used.

# Assumptions

Rovers should not go outside of plateau area.

Error handling: many error handling situations should usually be business requirements. In the description of rovers problem, there is not much information about how to handle error situations such as bad rover instructions “K” instead of “L”, “R” and “M”. So I just throw exception for now.

# Run the program

Starting point of the program is **RoverController.cs**