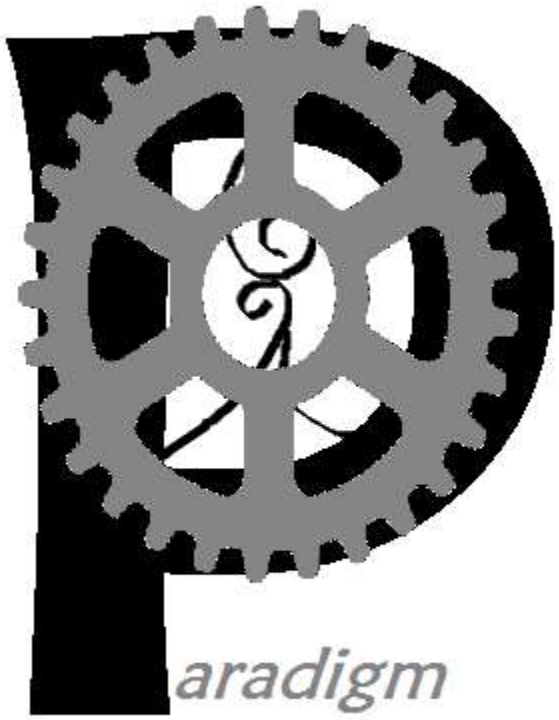
Test Suite User Manual



Paradigm
Damien Moeller
Aimee Phillips
Cavaughn Browne
Christian Norfleet

Table of Contents

Test Suite Overview

The purpose of the Test Suite is to create appropriate test data for early undergraduate computer science student's programs. The test suite is to be run concurrently with the program being tested [the test suite does not run the program].

The Test Suite allows for a variety of test settings that can be saved for later use. After running the test suite can then save a report which can be reviewed by the student's professor to help validate the program. During testing, if the student hand-writes expected results, they may compare them to the results, which is given by the program, and analyzed within the Test Suite.

The user manual is designed to explain the different functions and features of the Test Suite.

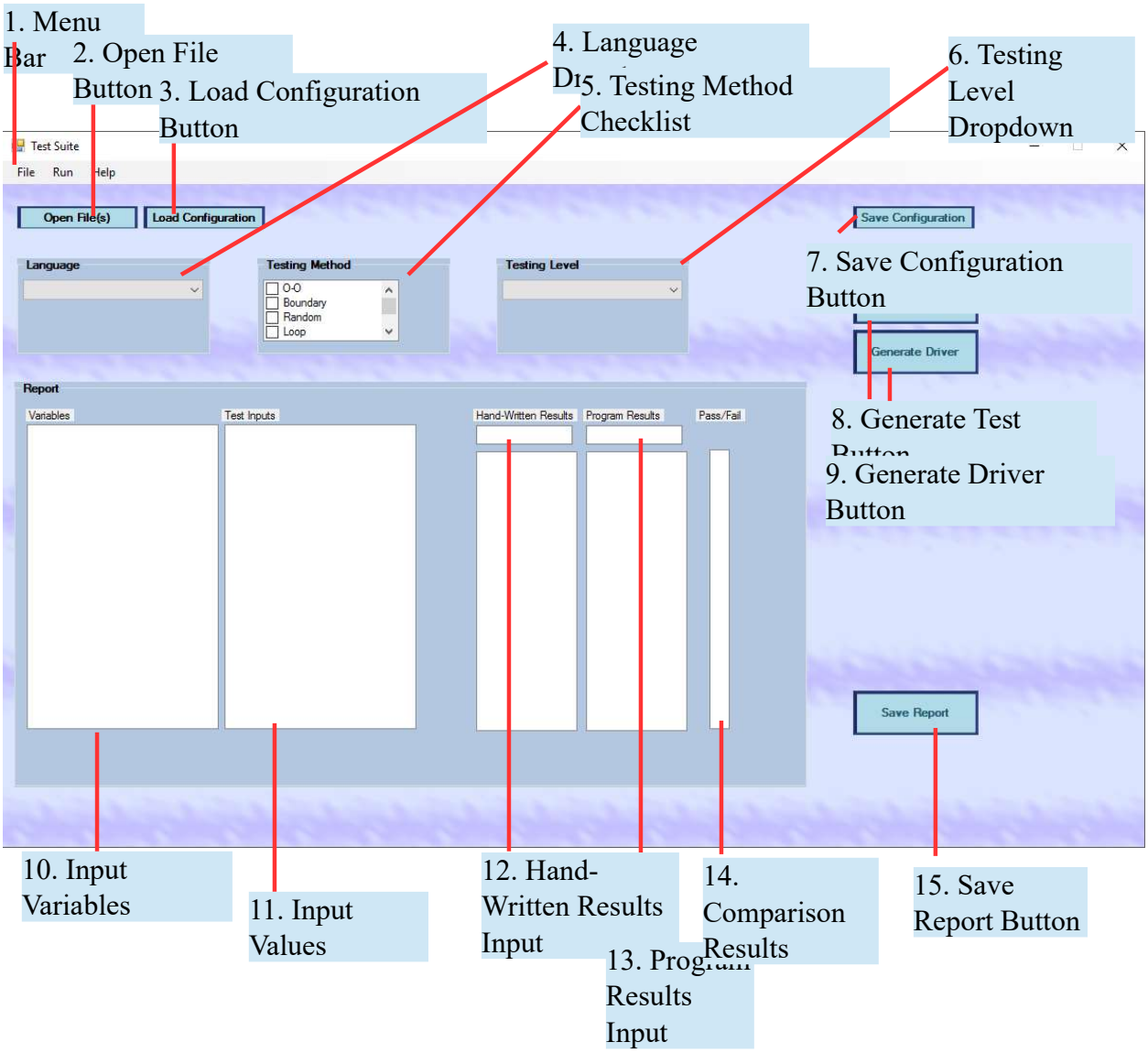
System Requirements

Windows OS, .NET Framework 4.5 and above, 128 MB RAM

Installation Instructions

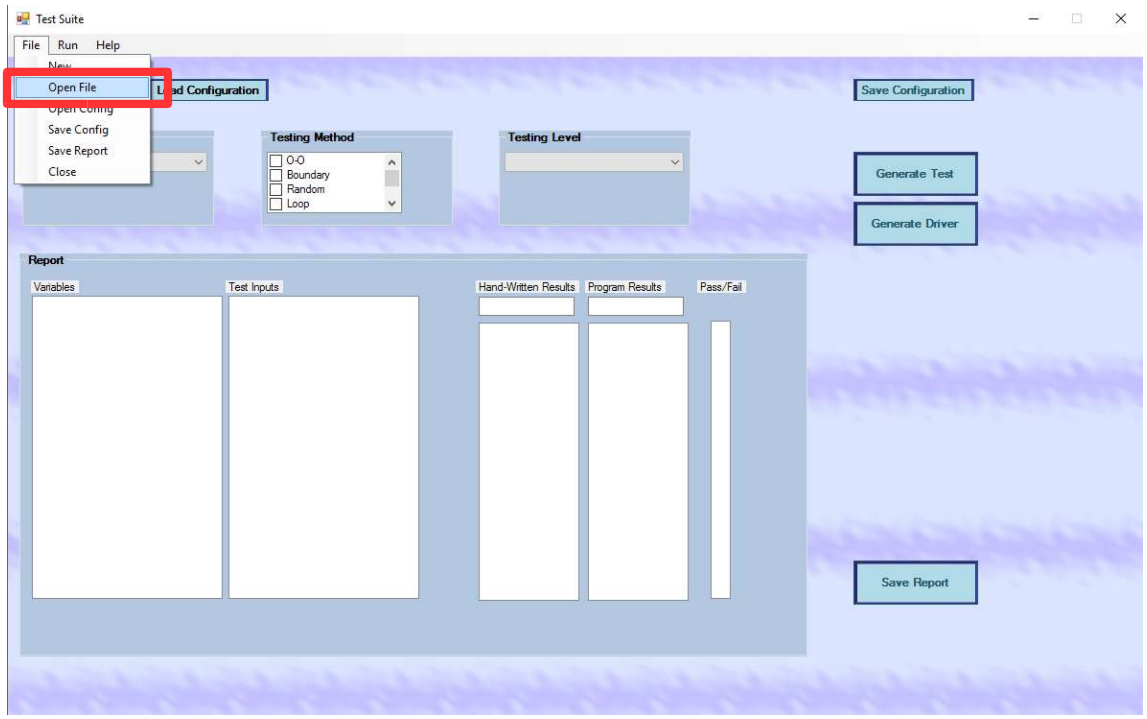
The installation of the Test Suite consists of simple drag and drop operation of the Test Suite folder onto your hard drive. If so desired, the Test Suite can just be used by double clicking the TestSuite.exe inside of the Test Suite folder on the from the File Explorer on the CD or external drive.

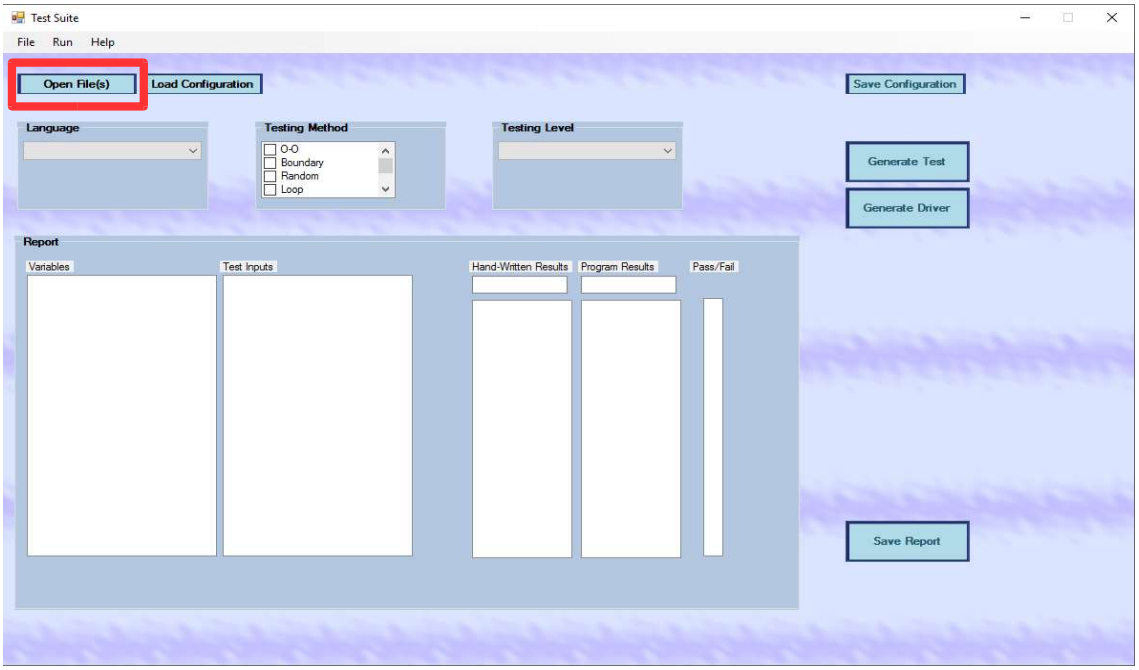
GUI


1. Menu Bar
2. Open File Button
3. Load Configuration Button
4. Language Dr
5. Testing Method Checklist
6. Testing Level Dropdown
Test Suite
File
Run
Help
Open File(s)
Load Configuration
Save Configuration
Language
Testing Method
O-O
Boundary
Random
Loop
Testing Level
7. Save Configuration Button
Generate Driver
Report
Variables
Test Inputs
Hand-Written Results
Program Results
Pass/Fail
8. Generate Test Button
9. Generate Driver Button
Save Report
10. Input Variables
11. Input Values
12. Hand-Written Results Input
13. Program Results Input
14. Comparison Results
15. Save Report Button

Opening Source Code

To open the .cpp and/or .h files you wish to test, either click File → Open Files or click the Open File(s) button.



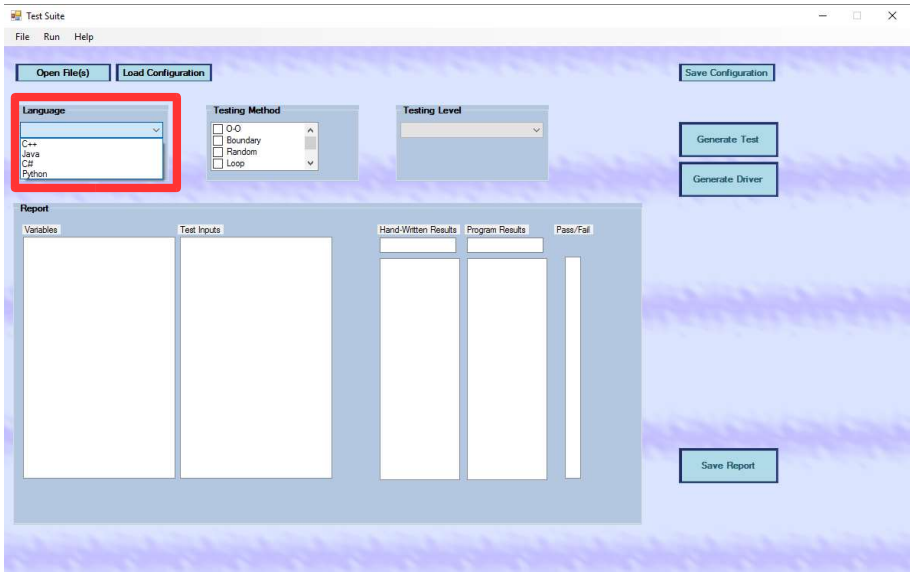


Setting Test Configuration

The configurations of a test must be specified by the user before a test can be generated from source code. Some of these configurations include the language being read, the testing method, and expected exception handling (or test level).

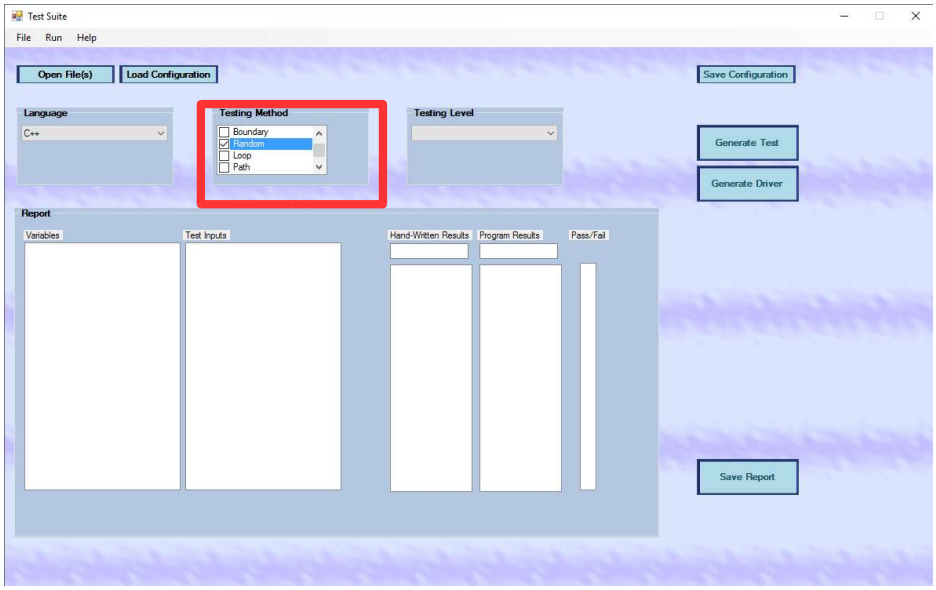
Language

You can set the language through the language drop-down list. (Version 1.0.0 only supports C++).



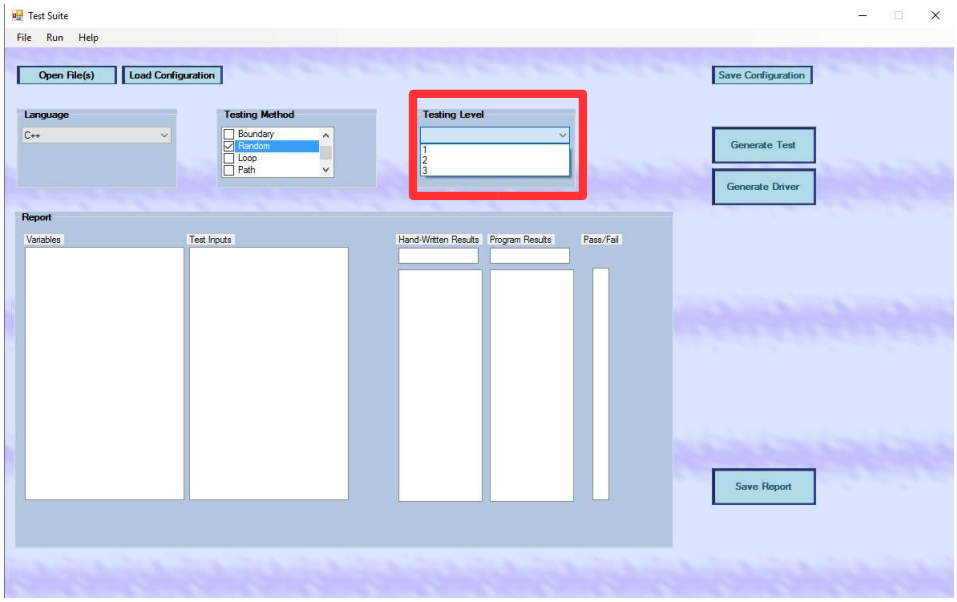
Testing Method

The Test Suite supports a number of different testing methods which can be selected from the Testing Method Checklist.



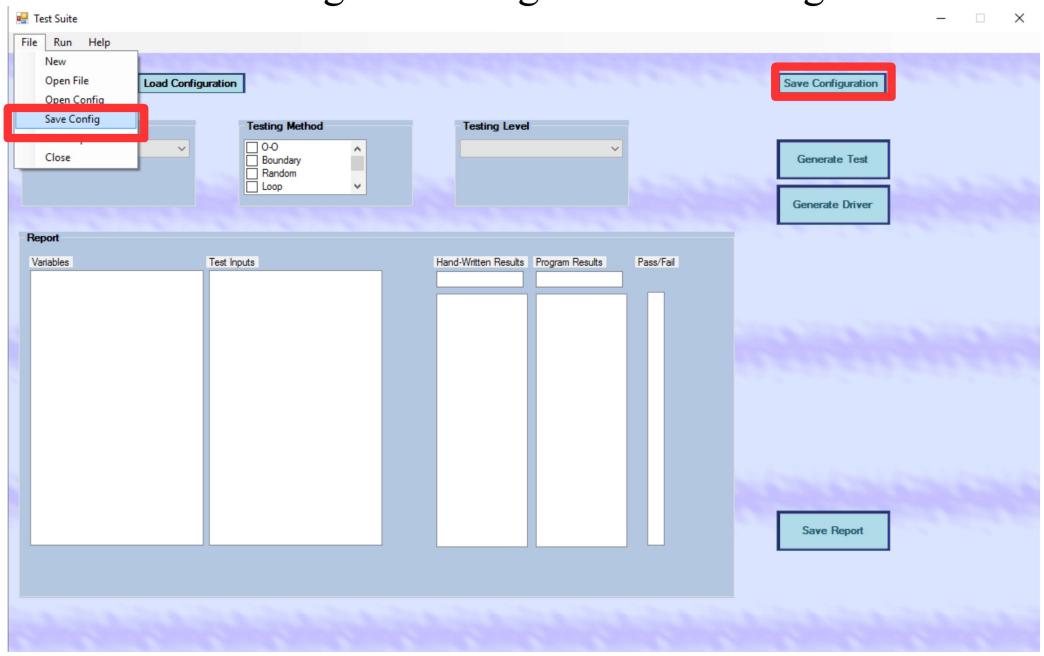
Testing Level

The testing level represents the degree of exception handling you wish to use during testing. The higher the testing level, the more exception handling the Test Suite will consider.



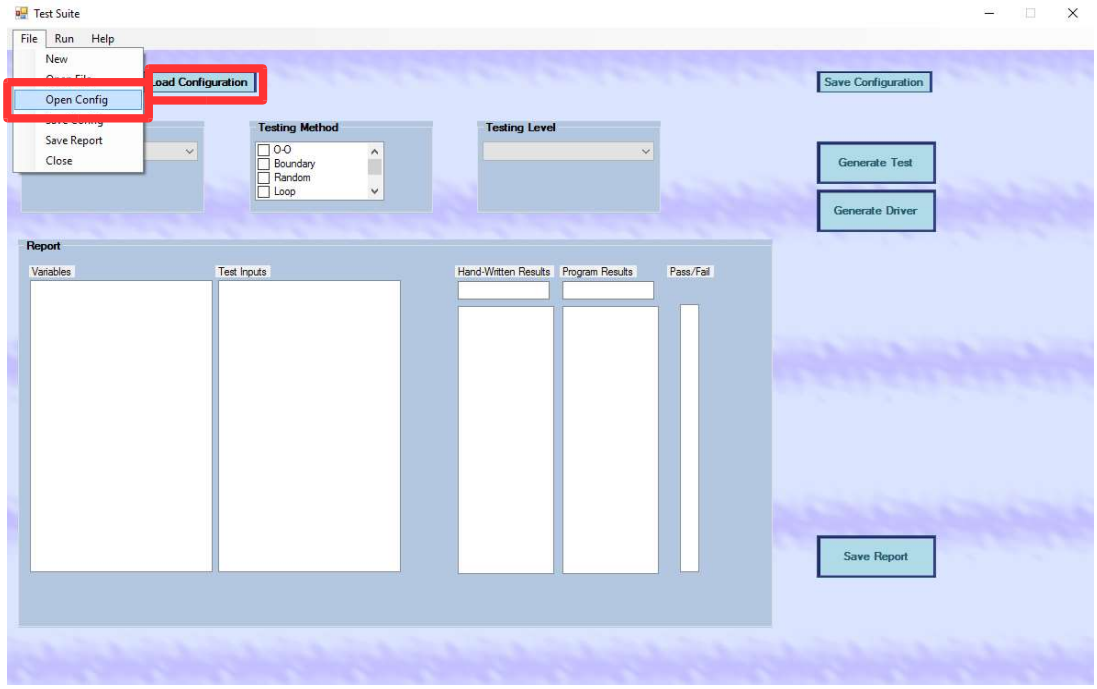
Saving a Configuration

After all of the test settings have been set, you may save the configuration to be used for later test. This is achieved by either clicking File → Save Config or clicking the Save Configuration button.



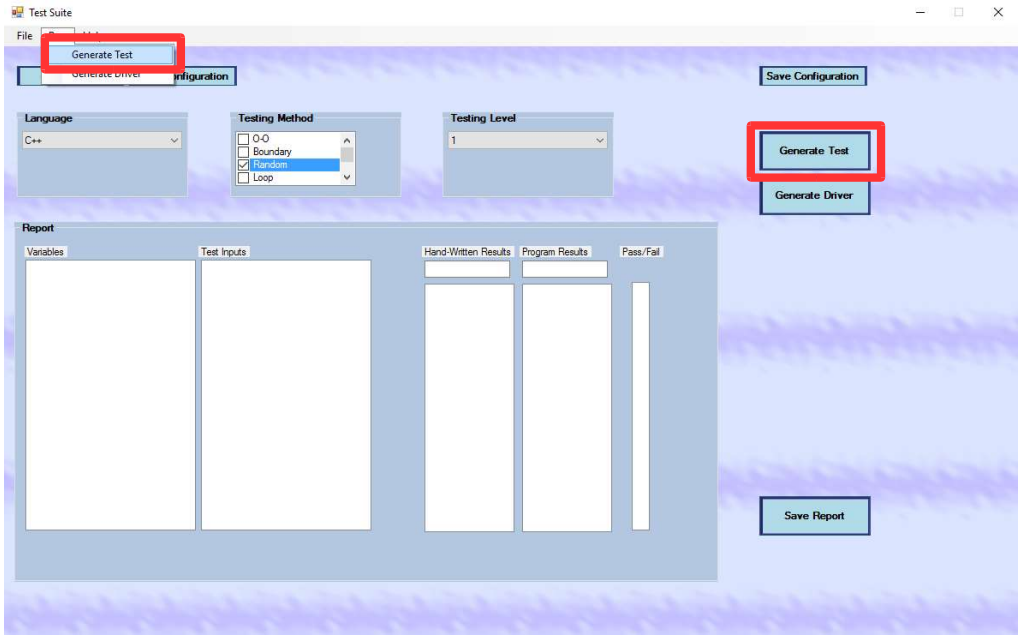
Loading a Configuration

You may load a configuration file if you have one saved. Once loaded, the settings within that file can be used to generate your test. You can load a configuration either by clicking File → Open Config or by clicking the Load Configuration button.

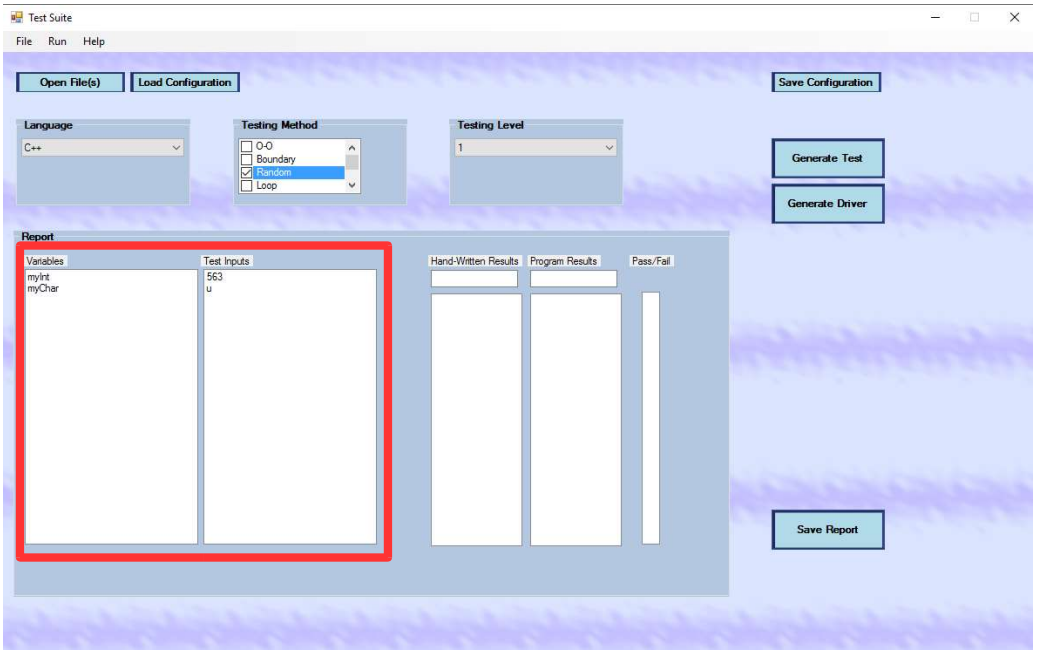


Generating Test Variables

Once you have the configuration set you may generate test data to test your source code. This can be done by clicking Run → Generate Test or clicking the Generate Test button.

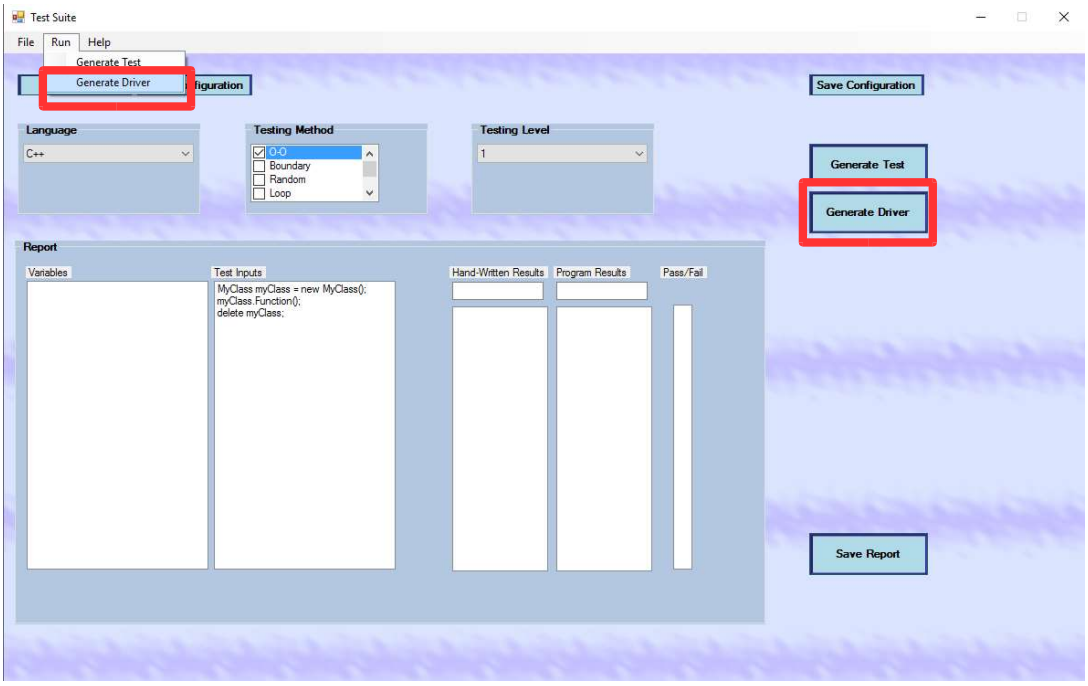


Once the test inputs are generated to use in your program, the variables and the values to be used for input are displayed in the Input Variables and Input Values boxes.

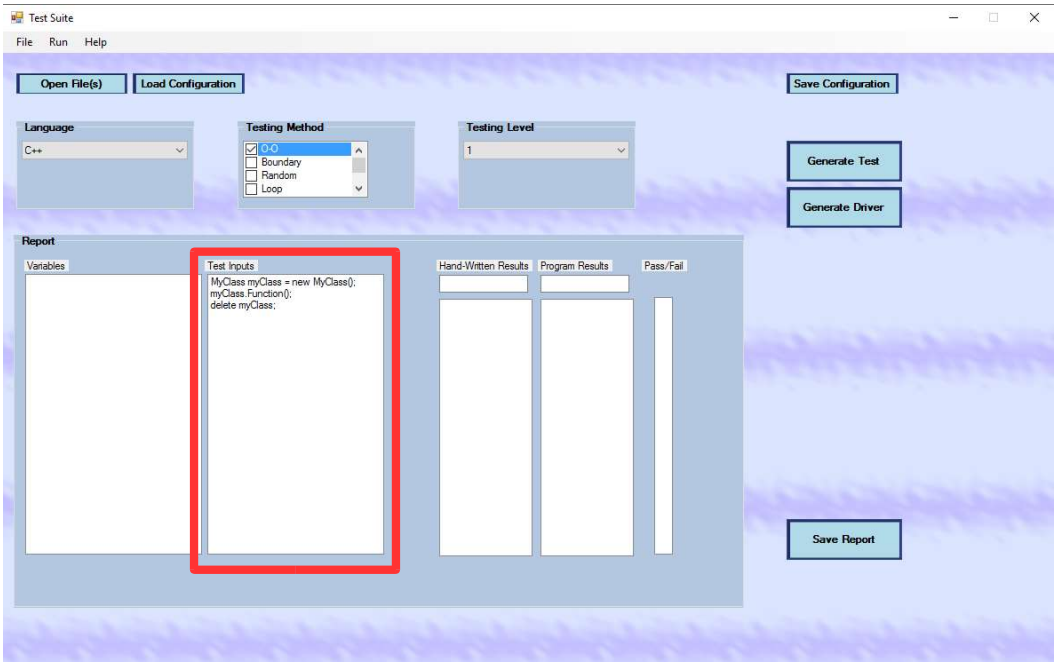


Generating Test Drivers

Once you have the configuration set you may generate test drivers. This can be done by clicking Run → Generate Driver or clicking the Generate Driver button.

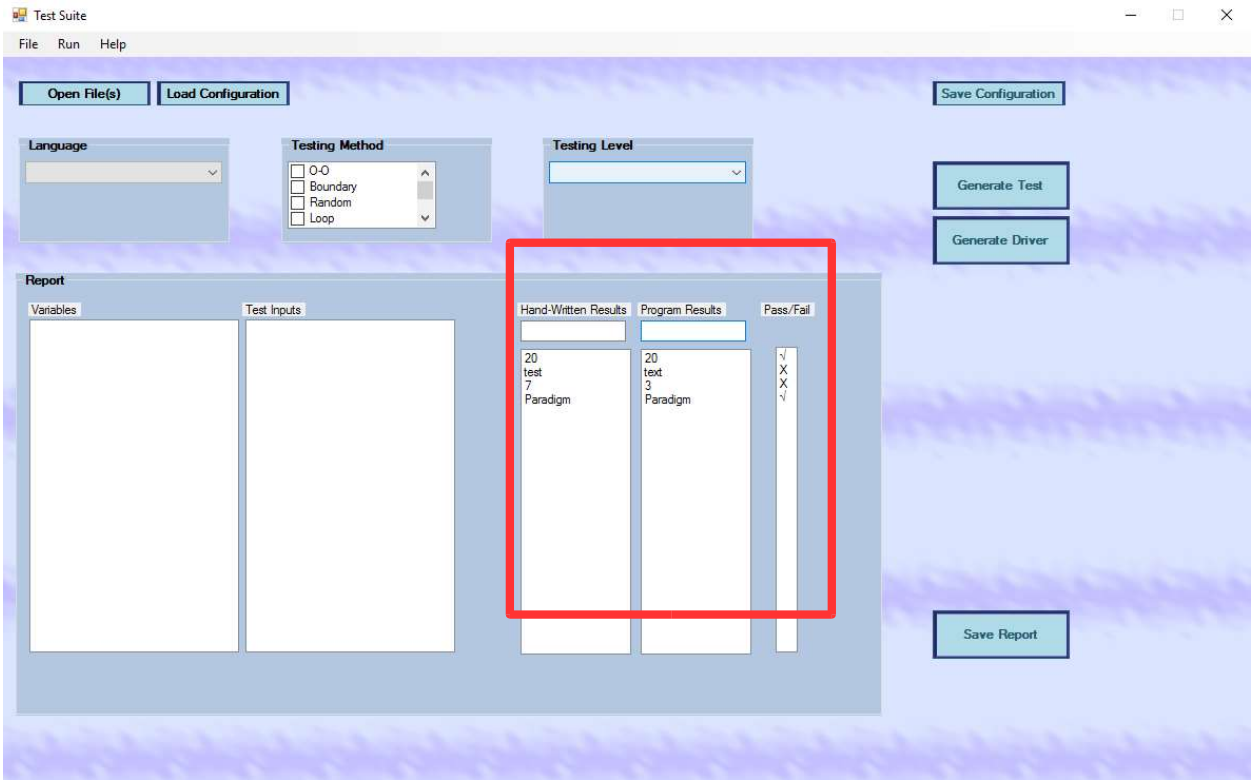


The test driver code will appear in the Test Inputs box which can then be copy pasted into a main program to test your class.



Comparing Results

At any point you may compare the results you expected with the results your program has produced. Type the results into the Handwritten Results Input or the Program Results Input box. The results of the comparison between the two will appear in the Comparison Results box. A √ will appear if the two results are equal and an X if they are unequal.



Saving a Report

After you have generated a test, you may save a report which includes the test inputs and any results you have typed into the program. The test and the results will be inserted into a template for readability and consistency. To save the report either click File → Save Report or click the Save Report button.