Test Report: Truss

Ting-Yu Wu

December 15, 2020

1 Revision History

Date	Version	Notes
December 11, 2020	1.0	Initial version of VnV report

2 Symbols, Abbreviations and Acronyms

symbol	description
T	Test

For the other symbols, abbreviations and acronyms, see SRS.

Contents

List of Tables

This document is a report on the results of a testing suite for Truss. Detailed descriptions of the tests executed can be found in VnV Plan

3 Functional Requirements Evaluation

All the functional requirements have been met.

4 Nonfunctional Requirements Evaluation

4.1 Accuracy and Verifiability

The outputs generated by Truss were compared to the solutions from https://skyciv.com/free-truss-calculator/ with the same input data.

4.2 Understandability

Questions	Answer
Consistent indentation and formatting style?	yes
Explicit identification of a coding standard?	yes
Are the code identifiers consistent, distinctive, and meaningful?	yes
Are constants (other than 0 and 1) hard-coded into the program?	no
Comments are clear, indicate what is being done, not how?	yes
Is the name/URL of any algorithms used mentioned?	yes
Parameters are in the same order for all functions?	yes
Is code modularized?	yes
Descriptive names of source code files?	yes
Is a design document provided?	no
Overall impression?	8

Table 1: Understandability grade sheet

4.3 Portability

The system can be performed on different operating systems with a terminal and python 3.8.

4.4 Maintainability

The completeness of the documents of multiple versions and issue tracking are well-organized on the Truss GitHub repo https://github.com/tingyuw/cas741.

4.5 Reliability

The software didn't break during the execution, and it takes less than one second to perform.

5 Comparison to Existing Implementation

The comparison will be done between Truss and Truss Calculator https://skyciv.com/free-truss-calculator/.

6 Unit Testing

The actual implementation of the unit tests can be found in VnV Plan section 6. All test cases are performed by test classes built with the help of Pytest. All tests succeed.

7 Changes Due to Testing

No changes have been done. However, a problem was found during the implementation. We can't let users decide both distances between joints and angles of truss memebers because some values can't properly position a truss structure. This problem should be solved in the future.

8 Automated Testing

Tools used for automated testing are mentioned in VnV Plan section 4.5. All system and unit tests passed. A Travis CI build for the code is in conjunction with Drasil. Here is the link to see the build https://travis-ci.com/github/tingyuw/Drasil. It is passed.

9 Trace to Requirements

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Table 2 shows the dependencies between the test cases and the requirements. Requirements can be found in SRS.

Requirements	**Test section**
R1	section 3
R2	section 3
R3	section 3
R4	section 3
R5	section 3
NFR1	section 4.1
NFR2	section 4.1
NFR3	section 4.2
NFR4	section 4.3
NFR5	section 4.4
NFR6	section 4.5

Table 2: Traceability Between Test Cases and Requirements

10 Trace to Modules

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Table 3

shows the dependencies between the test cases and the modules.

Requirements	**Test section**
Input Parameters Module	3, 4.1, 4.2, 4.4
Input Constraints Module	3, 4.1, 4.2, 4.4
Calculation Module	3, 4.1, 4.2, 4.4
Output Module	3, 4.1, 4.2, 4.4

Table 3: Traceability Between Test Cases and modules

11 Code Coverage Metrics

The following code coverage of each module is measured by Coverage.py.

Modules	**Code coverage**
Calculations.py	100%
Constants.py	100%
Control.py	100%
InputConstraints.py	15%
InputParameters.py	100%
OutputFormat.py	76%

Table 4: Code coverage of each module

Note that the purpose of the InputConstraints.py is to verify the inputs and display an error message if any component is out of bounds. Therefore, the code coverage of this module is low for valid input sets.

References

W Spencer Smith, Zheng Zeng, and Jacques Carette. Seismology software: State of the practice. *Journal of Seismology*, 22(3):755–788, 2018.

Ting-Yu Wu. SRS. https://github.com/tingyuw/cas741/blob/master/docs/SRS/SRS.pdf, 2020a.

Ting-Yu Wu. VnV Plan. https://github.com/tingyuw/cas741/blob/master/docs/VnVPlan/VnVPlan.pdf, 2020b.