

***CSC 413 Project Documentation***  
***Fall 2019***

***Corey Russ***

***917717871***

***CSC 413-03***

**[https://github.com/csc413-03-  
fall2019/csc413-p2-scrable](https://github.com/csc413-03-fall2019/csc413-p2-scrable)**

## Table of Contents

1	Introduction .....	3
1.1	Project Overview .....	3
1.2	Technical Overview .....	3
1.3	Summary of Work Completed .....	3
2	Development Environment.....	3
3	How to Build/Import your Project .....	3
4	How to Run your Project.....	3
5	Assumption Made .....	3
6	Implementation Discussion.....	3
6.1	Class Diagram .....	4
7	Project Reflection.....	5
8	Project Conclusion/Results .....	5

# 1 Introduction

## 1.1 Project Overview

The purpose of this project is to create code which can be applied to certain kinds of text files. These text files contain instructions on how the program should operate.

## 1.2 Technical Overview

The purpose of this project is to create the code that drives a program. The program reads a source file containing certain labels which correspond to certain actions. These actions are then applied to actual source code which run based on how the labels operate. The operations of each label do not change and only the source file will change. This allows the code to be applied to any source code given we have a source file to drive the program.

## 1.3 Summary of Work Completed

In order to get the program skeleton to function, I implemented several ByteCode classes which each represent their individual instructions. Each bytecode is abstracted from a general ByteCode class which allows us to use polymorphism with functions and make for better functionality and easier readability of code. A virtual machine class essentially runs these bytecodes. A stack and an arraylist are used to keep track of bytecodes being loaded from a text file.

# 2 Development Environment

- a. Version of Java used: 12.0.2
- b. IDE used: IntelliJ IDEA Ultimate 2019.2.1

# 3 How to Build/Import your Project

To import the project, open IntelliJ and create a new project from existing sources. All fields can be left as default. Select the root of the project folder as the source folder for the project.

# 4 How to Run your Project

First, we need to link a source file to the project. Edit configurations by clicking Run -> Edit Configurations. In the program arguments section, enter the path for the text source file. Then, through IntelliJ, right click the Interpreter file and click run interpreter.main(). In the console that appears, enter an integer and press enter. The result will be displayed in the console.

# 5 Assumption Made

One assumption made is that the source .cod files contain no errors. This allows the bytecodes to be read correctly and run perfectly. Also, we assume that the actual code for running the project is correct as well.

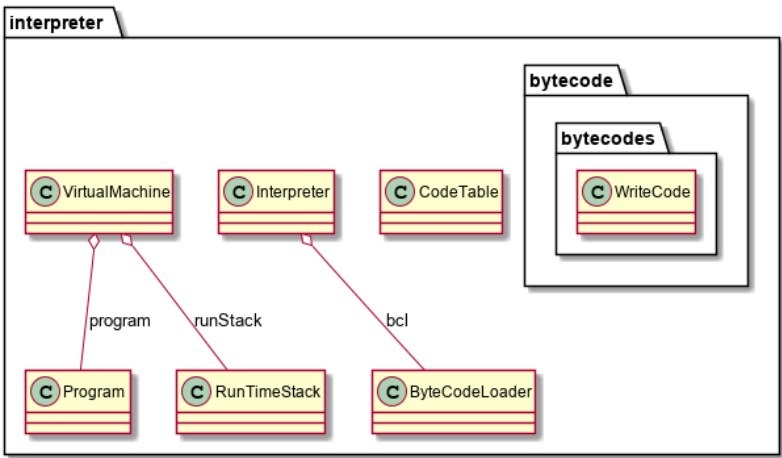
# 6 Implementation Discussion

By utilizing polymorphism we are able to easily create an abstraction which allows the bytecodes to run using identical code. The project utilizes several ArrayLists and Stacks to represent and store

bytecodes. A program class is used to hold an entire program consisting of these bytecodes. A virtual machine class is used to run the bytecodes.

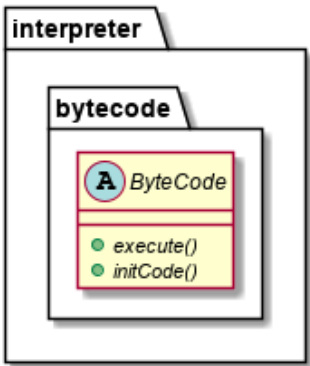
6.1 Class Diagram

CSC413-P2-SCRABLE's Class Diagram



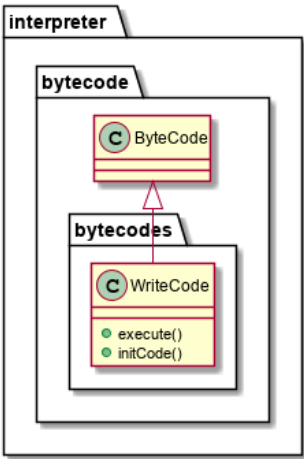
PlantUML diagram generated by SketchIt! (<https://bitbucket.org/pmesmeur/sketch.it>)  
For more information about this tool, please contact philippe.mesmeur@gmail.com

BYTECODE's Class Diagram



PlantUML diagram generated by SketchIt! (<https://bitbucket.org/pmesmeur/sketch.it>)  
For more information about this tool, please contact philippe.mesmeur@gmail.com

BYTECODES's Class Diagram



PlantUML diagram generated by SketchIt! (<https://bitbucket.org/pmesmeur/sketch.it>)  
For more information about this tool, please contact philippe.mesmeur@gmail.com

## 7 Project Reflection

I regret not using time closer to when the project was assigned. Furthermore, I did not fully understand the project when I began to work on it. This caused me to redesign my project as I was working on it, which was not helpful and wasted time.

## 8 Project Conclusion/Results

In conclusion, this project turned out to be more difficult than I originally anticipated. It was really frustrating to deal with. Handling all the various bytecodes and keeping the design straight proved to be difficult. If I started the project earlier and spent a little more time, I could have been more streamlined and finished earlier.