**Team Project 3**

Project 3B – Morse Code

**Luke Janis**

**Montana Shaw**

**Nunzio Lamartina**

**Robert Blocker**

**March 12th, 2022**

**Table of Contents**

[1.](#_gjdgxs) System Design 2

[2.](#_30j0zll) UML Diagram 2

[3.](#_1fob9te) Test Cases 2

[4.](#_2et92p0) Team Member Contribution 2

[5.](#_tyjcwt) Possible Improvements 2

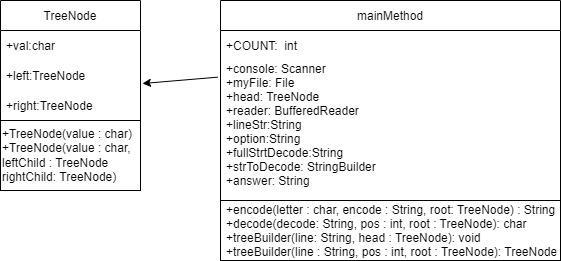
# System Design

The program consists of two classes. The TreeNode class and the main class.

**TreeNode**- Consists of values and constructors necessary to build a binary tree from the Morse\_Code text file.

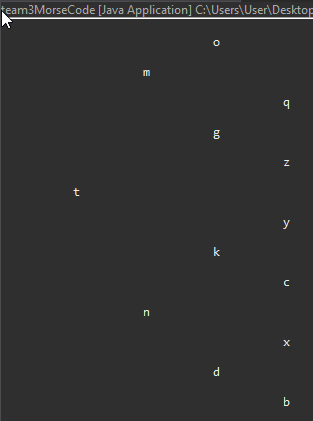
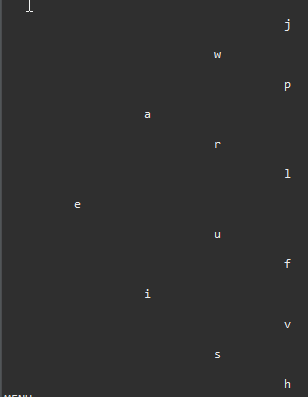
**team3MorseCode (main class)**- Consists of a decode method that recursively checks the binary tree for valid combinations of dots and dashes. The method then replaces the valid morse code with its corresponding letter. The class also contains the encode method that takes letters in the binary tree and changes them to their corresponding morse code. The last method of the class is the treeBuilder method that constructs a binary tree of morse code.

# UML Diagram



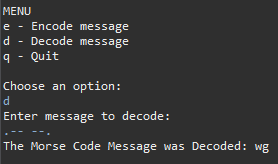
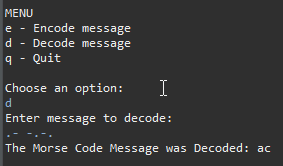
# Test Cases

Here, we created and printed out the tree for the Project. On the terminal the output is vertical.



The Left Side of the Tree The Right Side of the Tree

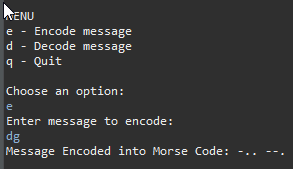
Let’s start to decode some Morse Code for our first tests.



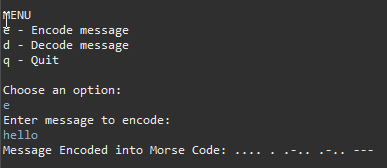
For the first test we choose to decode and enter “.- -.-.”. We expect the message to be “ac” and after hitting enter, we see the expected output confirmed.

For the second test we choose to decode and enter “.-- --.”. We expect the message to be “wg” and after hitting enter, we see the expected output confirmed.

For our next tests, we want to encode our message into Morse Code.



For the first encoding test we choose to encode by typing “e” and then entering “dg”. We expect the morse code to be “-.. --.” and after hitting enter, we see the expected output confirmed.



For the next encoding test, we want to encode the message “hello”. After entering the message, we expect the Morse Code to be “.... . .-.. .-.. ---”. After running the test, we see our program output the correct Morse Code.

# Team Member Contribution

Luke Janis – Luke worked on the development of the code for the project. Through logic and testing, Luke helped make sure the code functioned properly. Luke helped develop code in the TreeNode class and in the main method. Luke helped with the logic behind creating an efficient algorithm and program.

Montana Shaw – Montana worked on the development of the code for the project and also collaborated with Robert to document the project. He helped to document the project logic during the group meetings. Also developed code to create a working, efficient program.

Nunzio Lamartina – Nunzio worked on the development of the code for the project. Nunzio helped create the project’s logic and worked on the Encoding and Decoding parts of the project in the main method class. Nunzio helped with creating an efficient algorithm while making sure the code functioned properly.

Robert Blocker – Robert worked on the development of the code for the project and also collaborated with Montana to document the project. Worked on the code development to decode and encode. Also worked on logic for an efficient algorithm. Robert helped to document the project logic during the group meetings.

# Possible Improvements

Our group took note of the possible improvements from the last two projects to make sure we had a program that was more conscious of the time complexities and had efficient code.

A possible improvement could be the readability of the Tree that we made and output to the terminal at the start of the project, however we see that it is correct and that is not a large problem for the project.