**Morse Code Parser**

Drew Donaldson

Alyssa Pettit

William Waugh

**System Design**

The program first reads in the Morse code text file and loads it into the binary Morse code tree ready to translate letters to Morse code, or Morse code to letters. Then, it asks the user to input a string to be translated, either Morse code or letters. Based on the input, it determines if Morse code needs to be translated to letters, or letters need to be translated to Morse code. If translating from Morse code, it individually looks at every dot and dash of each letter, and traverses the Morse code tree by advancing left if dot, and right if dash then returns the node located. If translating from letters, it uses a depth first search through looking for a letter match then returns the dots and dashes representing the letter based on the path taken to find the letter.

**Contributions**

Everyone worked on the project together simultaneously via screenshare and GitHub and reviewed and edited the code as a group. All documents were created together and edited via Discord chat.

**Improvements**

The program currently only provides a translation based on if the string contains a dot or a dash or not. However, if the characters are a mix of letters and Morse code symbols, then it will return an incorrect response attempting to translate from Morse code or will return nothing. An improvement on our program would be to individually handle every letter or Morse code symbol and produce a mixed translation response. A simple solution would be that if it cannot find what it is looking for with one method, to attempt to switch to the other and see if a result is found.

**Test Cases**

1. Entered: mississippi

The program produced the code: -- .. ... ... .. ... ... .. .--. .--. .. as expected.

1. Entered: -- --- .-. ... .

The program produced the word: morse as expected.

**UML Diagram**

