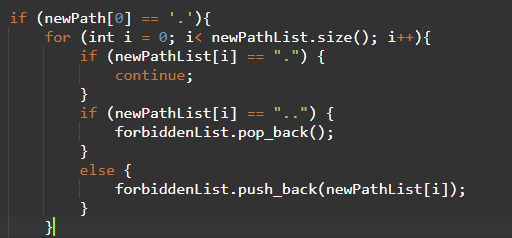
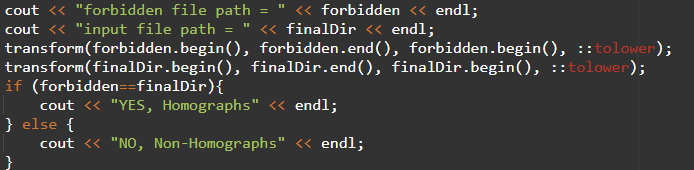
There are several parts of this lab that are key in the Homograph detecting area of this project. For example the code below will search the inputted text and determine if it contains a “.” because that would mean that the input is not going back to the home directory. Otherwise the code will take the input and add it to the current directory or go back a directory if the input has “..”. If there is just a single “.” added in then the code will ignore the input.



Another important part of the code for this lab would be the following, when a Homograph has been found and meets all criteria to be classes as a Homograph rather than a Non-Homograph the system will output the answer. This section of code will also determine if the input will be classes as one of the two options.



For this study there were two different cases that were tested, for the first series of tests Non-Homographs were tested. As shown below the code was tested in multiple ways in order to determine that the code could determine if an input was a Homograph or was a Non-Homograph.

(empty string)



E:\Users\Reynolds\downloads\secrets\secret.txt

Graphical user interface, text

Description automatically generated

\..\..\documents\secrets\secret.txt

Text

Description automatically generated

.\secrets\secret.txt

Text

Description automatically generated

c:\..\..\secret\secret.txt

Text

Description automatically generated

.\

Text

Description automatically generated

\\printershare\user\secrets

Text

Description automatically generated

The second part of the test on this code was to determine if it could recognise a Homograph and show such in the output. This is shown with the examples below, where several inputs were able to get in when already in the correct folder. In these examples the user would expect to be able to get to the requested file path but would be unable to since it is not exactly the correct input.

secret.txt

Text

Description automatically generated

..\..\..\reynold\downloads\.\secrets\secret.txt

Text

Description automatically generated

C:\Users\Reynold\downloads\secrets\secret.txt

Text

Description automatically generated

c:\users\reynold\downloads\secrets\secret.txt

Text

Description automatically generated

..\..\downloads\secrets\secret.txt

Text

Description automatically generated

..\..\downloads\..\downloads\secrets\secret.txt

Text

Description automatically generated

.\.\.\.\.\secret.txt

Text

Description automatically generated

In conclusion the code and tests that have been completed are enough to prove that the theoretical framework that was described in the test book can be applied and created. With what was presented above the code that was developed is enough to determine if the input file path is considered one of the following, a Homograph or a Non-Homograph. Those of which can be described as, two words that visually appear the same but consist of different characters. This is shown several times by the examples given above.