Cameron Thorp

CS210 Programming Languages

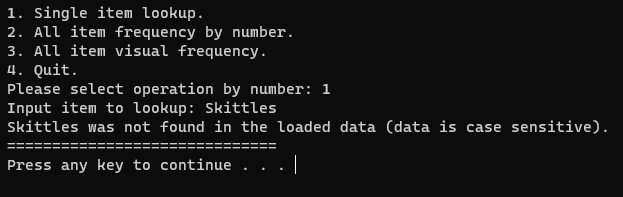
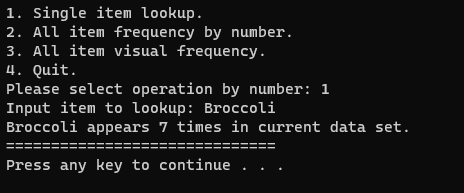
Project 3 Documentation

Overview:

The application created uses a single version of the input provided to allow for easy and accurate accessibility of data. Currently the input file is targeted via explicit pathing, however there is a constructor available for the data object which can allow a future feature implementation using relative pathing with user input. The user interface is meant to be simplistic and easy to use while gracefully handling invalid inputs from the user.

Customer Requirement 1:

This function accesses the data loaded from the input file and will return the frequency or a notification that the item is not present in the currently loaded data.



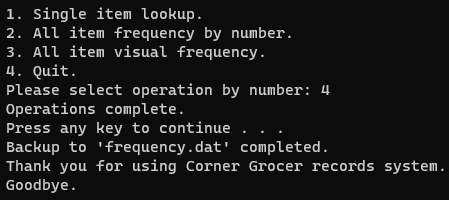
Customer Requirement 2 & 3:

These functions access the full set of input data and iterates over each item, returning the name of the item and its frequency to the console. Req. 2 returns numerical frequency, req. 3 returns visual frequency.



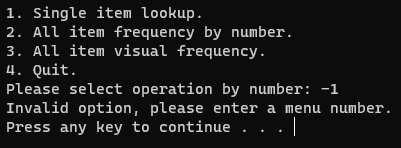
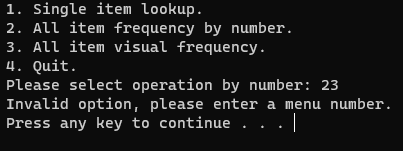
Customer Requirement 4:

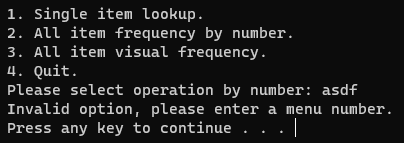
Upon selecting the exit option, the menu functionality is ended, the data set is written to a file (currently called frequency.dat) and the user is notified of this backup and thanked for using the software.



Optional Customer Requirement:

Users are unable to input a menu option that causes the program to produce an error. This is accomplished with a try catch block for each user input, these inputs being limited to the menu and single lookup functions.





Possible Program Enhancements:

In the future, this project could be improved by implementing relative file pathing, creating an executable to reduce compilation resource consumption, and by building a more robust UI in another language and utilizing the existing C++ functionality for the backend.