# **Documentation for the CornerGrocerItemTracker Project**

### **General Design of Code:**

For this project, I wrote a class named **ItemTracker**. This class can read the input file and analyze the items bought. The class contains various maps (std::map) that depict the name of a particular item and the amount it was bought not less than (an integer) times.

Let me give a brief overview of the methods defined in the class:

* **readFile()**: A method that wishes to open a specific text file, CS210\_Project\_Three\_Input\_File.txt, and analyze how often an item occurs. The lower edge of the page shows the error dialog in case the file does not open correctly.
* **getItemFrequency()**: Most users are unable to know how many times a specific item appears on the file, and this method resolves this problem. The input is given as the name of an item that has been sold. If the item was not sold, then such a method will return nothing.
* **displayAllItems()**: This method shows all the items posted for sale, including when they were sold.
* **displayHistogram()**: Each item sold on a given day accumulates one more up to the last item ever sold, and this item gets represented on a chart; no explanation is needed. In this way, the method allows the building of a chart that displays the sales of various products and paints them with unlimited stars. The number of stars is directly proportional to the amount of goods sold.
* **saveData()**: The necessary statistical data relevant to the number of items sold are saved to a file, frequency.dat. This file is handy when you are unsure about your memory.

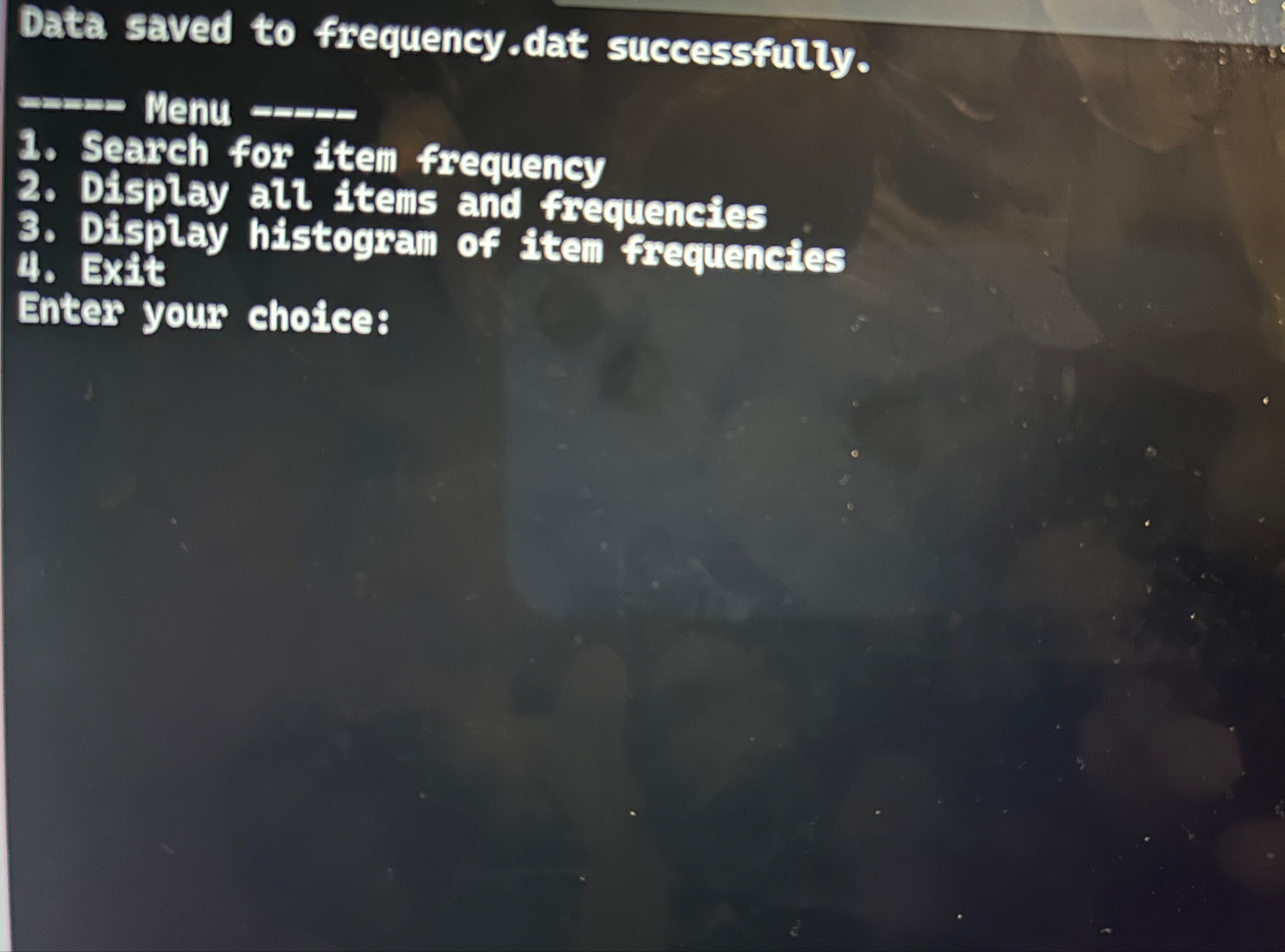
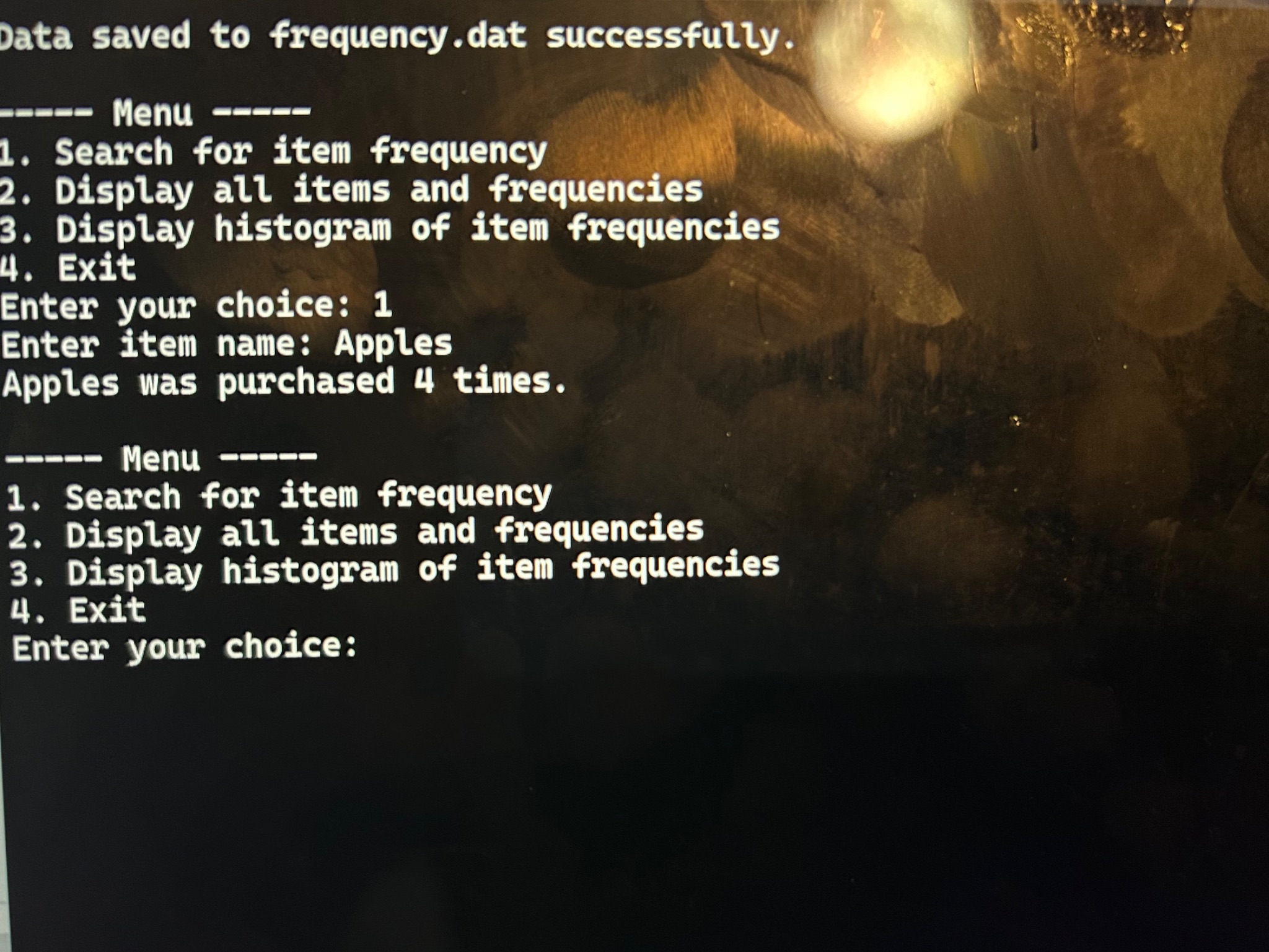
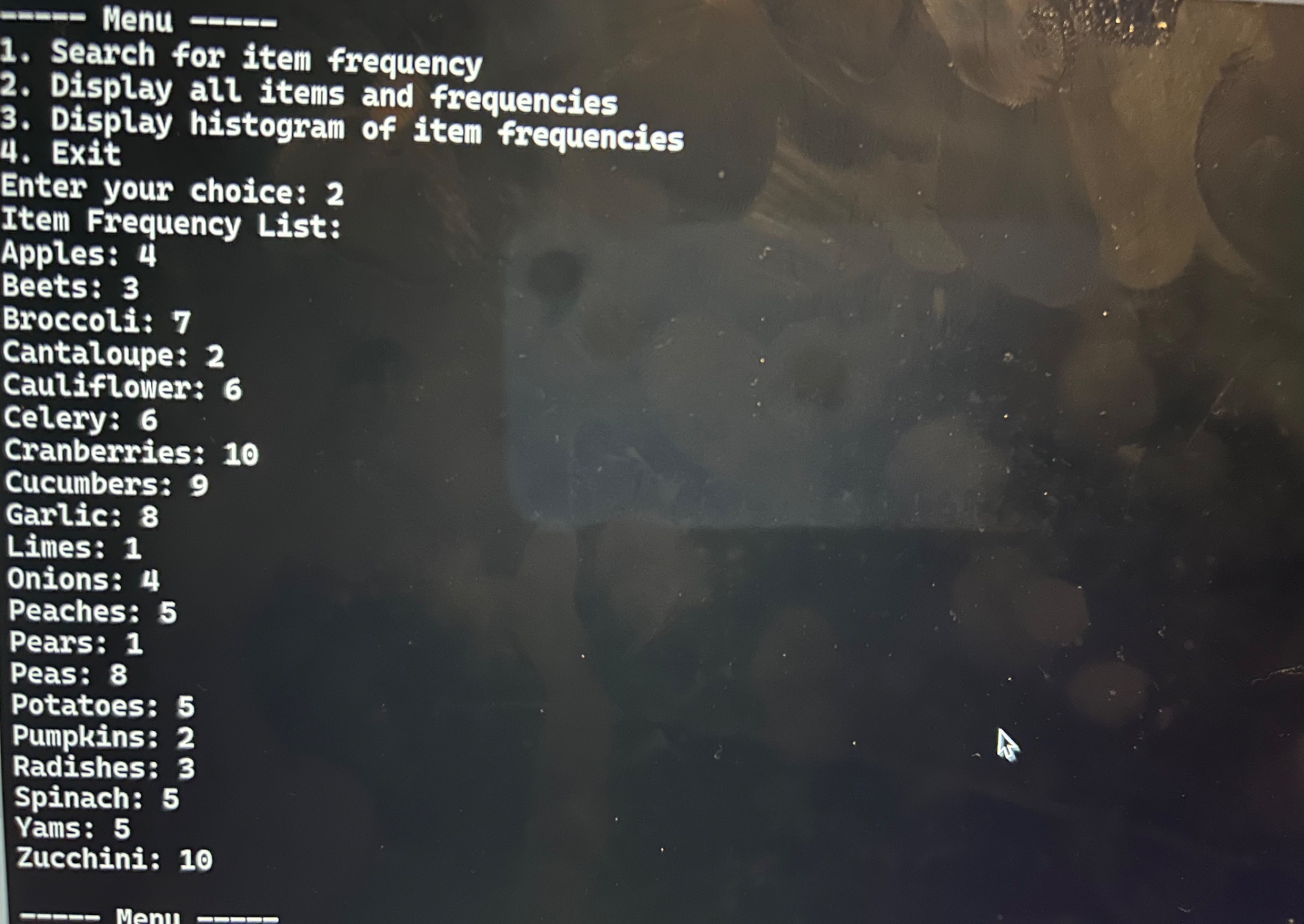
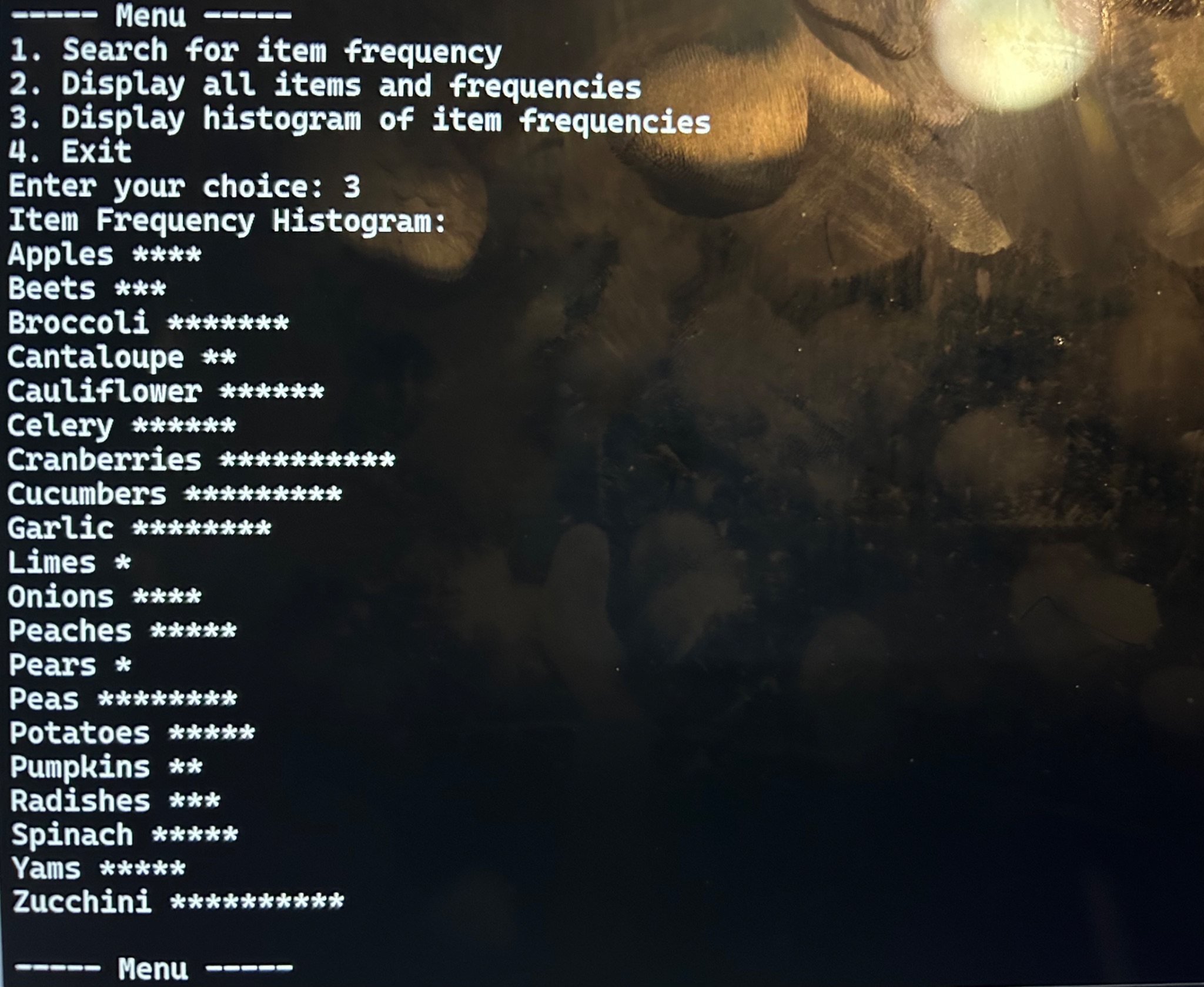
### **Menu Options:**

The task of the menu is to pick up the action that the user desires without much hustle. Each of the options does the following:

* **Search for Item Frequency**: Once you press this option, the program first asks for an item to be entered. Then, it tells you how many sales of that item have been recorded.
* **Display All Items and Frequencies**: This option returns a report volume containing the names of all items and the number of copies sold.
* **Display Histogram of Item Frequencies**: Other than that, the program will display the histogram with asterisks showing how many pieces of a particular item have been purchased.
* **Exit**: This option winds up the program.

### **Screenshots:**

This section contains the screenshots of the given program while in use:

* **Menu Display**: 
* **Search for Item Frequency**:
* **Display All Items and Frequencies**: 
* **Histogram Display**: 

### 

### 

### **APA Citations (if applicable):**

I do not have any outside sources to refer to in this project. Every line of code and program flow was an intellectual effort on my part. If any references were made, this information would be highlighted and placed here.

In case there is a reference I would like to make and how I would do so:

* ZyBooks. (2024). *CS210: Programming in C++*. All sources of information should be placed after this text. Available at: <https://learn.zybooks.com/>