

Corner Grocer Item Frequency Tracker

CS-210: Programming Languages

Erica Kinch | 12/14/2025

Program Purpose

The Corner Grocer Item Frequency Tracker was developed to analyze daily grocery transaction data and report the frequency with which each item is purchased. By converting raw transaction records into structured frequency summaries, the program supports data-driven decision-making related to store layout, product placement, and inventory management.

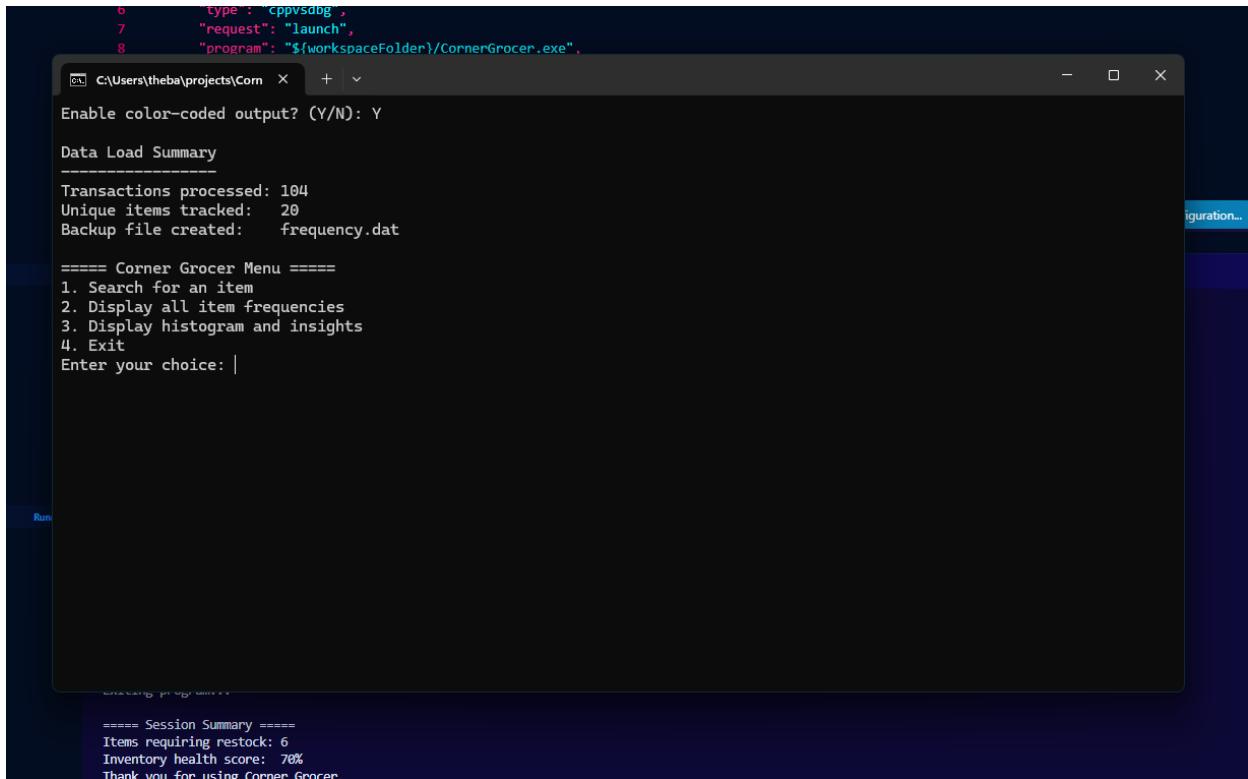
This project demonstrates foundational C++ programming concepts, including object-oriented design, file input and output, input validation, and effective use of the C++ Standard Library.

Program Design

The application is designed using object-oriented principles and centers around a dedicated `GroceryTracker` class. This class encapsulates all data-processing responsibilities, including reading input data, tracking item frequencies, generating backup files, and producing formatted reports. Separating data logic from user interaction improves code readability, maintainability, and scalability.

A `std::map<std::string, int>` container is used to associate grocery item names with their purchase frequencies. This data structure was chosen because it ensures unique keys, allows efficient lookups, and automatically maintains alphabetical ordering for consistent output. User input is

normalized to lowercase to support case-insensitive item searches, improving usability without altering the underlying data structure.



The screenshot shows a terminal window titled 'C:\Users\theba\projects\CornerGrocer'. It displays the following text:

```
6      "type": "cppvsdbg",
7      "request": "launch",
8      "program": "${workspaceFolder}/CornerGrocer.exe",
ON C:\Users\theba\projects\Corner x + v
Enable color-coded output? (Y/N): Y
Data Load Summary
-----
Transactions processed: 104
Unique items tracked: 20
Backup file created: frequency.dat

===== Corner Grocer Menu =====
1. Search for an item
2. Display all item frequencies
3. Display histogram and insights
4. Exit
Enter your choice: |

Run
-----
```

Session Summary

```
===== Session Summary =====
Items requiring restock: 6
Inventory health score: 70%
Thank you for using Corner Grocer.
```

Figure 1 Program startup showing successful input file loading and data validation summary.

Menu Options

The program provides a menu-driven interface that allows users to interact with the transaction data in several ways:

1. **Look up item frequency** – Displays how many times a specific grocery item appears in the input file.
2. **Display all item frequencies** – Outputs a complete list of grocery items and their purchase counts.

3. **Display histogram and insights** – Visualizes item frequencies using a text-based histogram and provides inventory insights such as top-selling items and demand indicators.

4. **Exit** – Safely terminates the program.

Input validation is implemented to ensure the program responds gracefully to invalid menu selections.

The screenshot shows a terminal window titled 'C:\Users\theba\projects\CornerGrocer'. The window displays a text-based histogram and various inventory insights. The histogram shows purchase frequencies for different items:

Item	Purchase Frequency	Category
apples	*****	[MED]
beets	***	[LOW]
broccoli	*****	[HIGH]
cantaloupe	**	[LOW]
cauliflower	*****	[HIGH]
celery	*****	[HIGH]
cranberries	*****	[HIGH]
cucumbers	*****	[HIGH]
garlic	*****	[HIGH]
limes	*	[LOW]
onions	****	[MED]
peaches	*****	[MED]
pears	*	[LOW]
peas	*****	[HIGH]
potatoes	*****	[MED]
pumpkins	**	[LOW]
radishes	***	[LOW]
spinach	*****	[MED]
yams	*****	[MED]
zucchini	*****	[HIGH]

Below the histogram, the program displays business insights, including top movers and inventory health:

==== Business Insights ====
Top Movers
Most Purchased: cranberries (10)
Least Purchased: limes (1)

Inventory Health
Inventory Health Score: 70% (Moderate)

Restock Recommendations

- beets (LOW stock)
- cantaloupe (LOW stock)
- limes (LOW stock)
- pears (LOW stock)
- pumpkins (LOW stock)
- radishes (LOW stock)

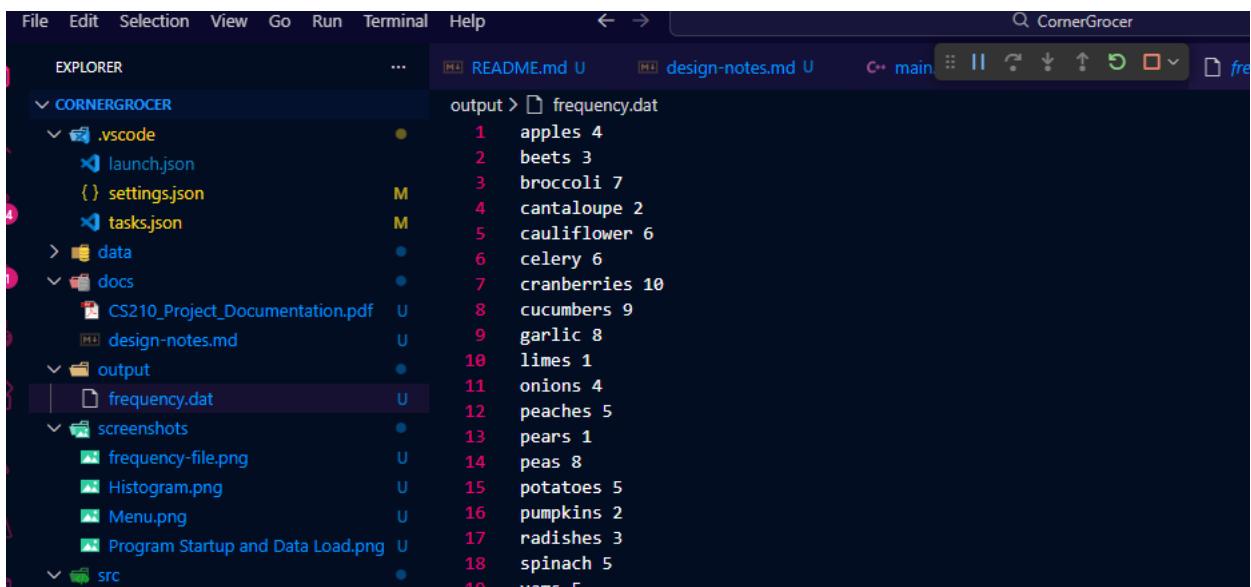
==== Corner Grocer Menu ====
1. Search for an item
2. Display all item frequencies
3. Display histogram and insights
4. Exit

Enter your choice: |

Figure 2 Text-based histogram and inventory insights generated from transaction data.

File Handling

The program reads grocery transaction data from an external text file, CS210_Project_Three_Input_File.txt, where each line represents a purchased item. After processing the data, the program automatically generates a backup file named frequency.dat that stores each item alongside its frequency count. This demonstrates proper use of file input/output operations, defensive programming, and persistent data storage.



The screenshot shows a dark-themed code editor interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled "CornerGrocer". Below the menu is a toolbar with icons for back, forward, search, and file operations. The left sidebar is titled "EXPLORER" and shows the project structure:

- CORNERGROCER**
 - .vscode
 - launch.json
 - settings.json
 - tasks.json
 - > data
 - docs
 - CS210_Project_Documentation.pdf
 - design-notes.md
 - output
 - frequency.dat
 - screenshots
 - frequency-file.png
 - Histogram.png
 - Menu.png
 - Program Startup and Data Load.png
 - src

output > frequency.dat

1	apples	4
2	beets	3
3	broccoli	7
4	cantaloupe	2
5	cauliflower	6
6	celery	6
7	cranberries	10
8	cucumbers	9
9	garlic	8
10	limes	1
11	onions	4
12	peaches	5
13	pears	1
14	peas	8
15	potatoes	5
16	pumpkins	2
17	radishes	3
18	spinach	5
19	vams	5

Figure 3 Automatically generated backup file (frequency.dat) containing item frequencies.

Summary

This project emphasizes core C++ programming skills, including class design, use of standard library containers, file handling, and input validation. It also highlights the importance of separation of concerns and clear program structure. The Corner Grocer Item Frequency Tracker

provides a practical example of transforming raw data into actionable insights through clean, maintainable C++ code.

```
===== Business Insights =====

Top Movers
-----
Most Purchased: cranberries (10)
Least Purchased: limes (1)

Inventory Health
-----
Inventory Health Score: 70% (Moderate)

Restock Recommendations
-----
- beets (LOW stock)
- cantaloupe (LOW stock)
- limes (LOW stock)
- pears (LOW stock)
- pumpkins (LOW stock)
- radishes (LOW stock)

===== Corner Grocer Menu =====
1. Search for an item
2. Display all item frequencies
3. Display histogram and insights
4. Exit
Enter your choice: 4

Exiting program...

===== Session Summary =====
Items requiring restock: 6
Inventory health score: 70%
Thank you for using Corner Grocer.
Recommendation: Prioritize restocking and prominent placement of HIGH-demand items to prevent stockouts.
ng shelf space for LOW-demand items.
Press any key to continue . . . |
```

Figure 4 Business insight output displaying top-selling and lowest-selling items, overall inventory health score, restock recommendations, and an end-of-session strategic summary to support data-driven layout and stocking decisions.