**CS 210 Project Three: Corner Grocer App**

Hector Banos Ramos 12/10/2023

The Corner Grocer App is designed to track items in a grocery store. It provides a menu-driven interface with options to search for an item, display item frequencies, print a histogram, and exit the program. The program reads item data from a file, processes it, and allows users to interact with the data through various menu options. The program also creates a *.dat* file containing each item at the beginning of the program execution.

**Pseudocode**

**START** program

DECLARE int userChoiceVar // Variable declaration

DECLARE bool exitRequest

DECLARE string fileName

// Dynamically allocate an instance of the InventoryItems class

CREATE InventoryItems object

// Call the function to read items from the file

CALL object.readItemsFromFile(fileName)

// Call the function to create a data file

CALL object.createDataFile("frequency.dat")

// Main program loop

WHILE NOT exitRequest:

// Print the menu options

CALL object.PrintMenu()

// Try block for error handling

TRY:

// Prompt the user for input and validate it

DISPLAY "Enter your choice: "

INPUT userChoiceVar

// Check for invalid input

IF userChoiceVar IS NOT an integer:

RAISE runtime\_error("Invalid input. Please enter a number."

// Check if the choice is within the valid range

IF userChoiceVar < 1 OR userChoiceVar > 4:

RAISE runtime\_error("Invalid choice. Please enter a number between 1 and 4.")

// Execute the selected menu option

CALL object.MainMenuOptions(userChoiceVar, exitRequest)

CATCH runtime\_error:

// Handle runtime error

DISPLAY "Error: " + ERROR\_MESSAGE

CLEAR the error flag

IGNORE invalid input

// Delete the allocated memory

DELETE object

**END** program

**Program Functionality**

Reading Items from File (readItemsFromFile): The program reads item data from a file specified by the fileName variable. Each line of the file is processed, assuming it contains one item, and updates the itemFrequencyMap in the InventoryItems class.

A screen shot of a computer code

Description automatically generated

Creating Data File (createDataFile): The program creates a data file named "frequency.dat" that stores the frequencies of each item. It iterates through the itemFrequencyMap and writes the item names and their frequencies to the file.

A screen shot of a computer program

Description automatically generated

Search for an Item (searchForItem): Users can search for a specific item by providing its name. The program uses the find function on the itemFrequencyMap to locate the item and displays its frequency if found.

A computer screen with text on it

Description automatically generated

Display All Frequencies (displayAllFrequencies): Users can choose to display frequencies of all items. The program iterates through the Map and prints each item along with its frequency.

A screen shot of a computer code

Description automatically generated

Print Histogram (printHistogram): Users can choose to print a text-based histogram of item frequencies. The program prints each item followed by a number of asterisks or any symbol corresponding to its frequency.

A screen shot of a computer code

Description automatically generated

Menu Printing (PrintMenu): The program prints a menu with options for searching, displaying frequencies, printing a histogram, and exiting the program.

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

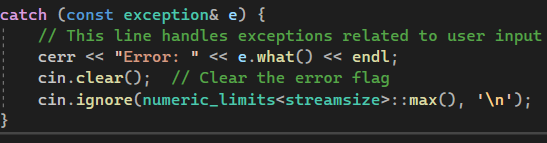
Description automatically generatedMenu Options (MainMenuOptions): The main function for handling user menu choices. Calls corresponding functions based on the user' s choice.

Input Validation (try-catch block in main): Utilizes a try-catch block to handle invalid user input gracefully. Ensures the user enters a valid menu choice and handles exceptions.

A black background with green and orange text

Description automatically generated

….



**Example of Program Execution**

Suppose we have the input file "CS210\_Project\_Three\_Input\_File.txt".

A screenshot of a computer

Description automatically generated

A black background with green and white text

Description automatically generatedA black background with green and white text

Description automatically generated 



A black background with white text

Description automatically generated

….

A black screen with white text

Description automatically generated

….

A black background with white text

Description automatically generated

1. The program reads items from the file.
2. The data file "frequency.dat" is created with the initial frequencies.
3. The user selects option 1 to search for an item (ex., " Onions").
4. The program displays the frequency of “ Onions"
5. The user selects option 2 to display all frequencies.
6. The program prints all items with their frequencies.
7. The user selects option 3 to print a histogram.
8. The program prints a histogram based on item frequencies.
9. The user can repeat these steps or choose option 4 to exit the program.