

Christian Roccanova

CS 162 Project2 design/reflections

4/26/2017

The purpose of Project 2 is to create a grocery list. It will need to take the name of the item and the unit it is sold in as strings and the unit price and the quantity to purchase as integers. A menu function will allow the user to choose to add an item, remove an item, display the current list and exit the program.

To do this, there will be 2 classes, List and Item. The Item class will contain set and get functions for each piece of item information the user inputs. Item will also contain a price function to calculate the total price for that item by taking unit price and quantity as parameter and multiplying them.

The List class will contain an addItem function which will call all of the set functions using an item object. This function will also increase the size of the array this information is stored in if it exceeds the initial capacity. To calculate the total price of the list, a totalPrice function will sum the prices of all items in the list and return the total. There will also be a displayList function to print all of this information. A removeItem function will also be added to delete an unwanted item from the list. There will also be a duplicate function which will contain an overloaded "==" operator to test if an item has already been added or not. If the item is found to exist, it will then ask if the user wishes to overwrite this item.

Input	Expected output	Observed Output	Fix
Enter non integer value into quantity	Prints error message and asks for a new input	Prints error message and asks for a new input	n/a
Enter non integer value into unit price	Prints error message and asks for a new input	Prints error message and asks for a new input	n/a
Enter invalid menu choice	Prints error message and asks for a new input	Prints error message and asks for a new input	n/a
Chooses to display list when it is empty	Returns to menu	Program crashed	Added if statement to check if anything has been added to the list and return the user to the menu if it is empty.
Chooses to remove when list is empty	Returns to menu	Program crashed	Added if statement to check if anything has been added to the list and return the user to the menu if it is empty.
Choose to add an item after 4 items have	Program increases the size of the array before	Exception thrown. Access violation in	Unable to fix by deadline, program

already been added	asking for user input as normal	writing location.	crashes if it is asked to go beyond the initial bounds of the list
Choose to add an item	Program asks for item information, then returns to menu.	Segmentation fault	Changed -> to . operator
Choose to display the list after adding items	Prints list of items and their information along with the total price.	Prints list of items and their information along with the total price.	n/a
Choose to add an existing item	Prints "that item is already on the list" then asks if the user would like to overwrite the item.	Prints "that item is already on the list" then asks if the user would like to overwrite the item.	n/a

The final program ended up in largely the same format as it was originally designed. The main file contained a menu function that allowed the user to choose to add or remove an item, or display the list. The add item option was altered, however, to allow for the overloaded "==" operator to compare user input before calling the addItem function to allow for a possible overwrite. Another issue I ran into was that my initial design did not account for the user attempting to display the list or remove an item when the list was empty. As a result, a counter was added to the main file to keep track of how many items were in the list. If the remove or display options were chosen when the counter was zero, the program would print that the list was empty and then return to the menu. Along the same lines, an input validation class was also added to provide functions to validate all integer inputs in the program. The greatest issue I came across was in increasing the size of the array. I had thought that my logic was sound, but I continue to get access violation errors when trying to go past the initial size of four. I unfortunately, was unable to fix this error by the deadline.