

(20 Points)

Shopping Cart

In homework # 2, we used dynamic arrays to implement a shopping cart class.

Keeping the interface almost unchanged, please change the implementation of a shopping cart class from dynamic arrays to a linked list. The new implementation must still support all previous operations of the shopping cart which are listed below.

Shopping Cart - Functionalities

```
addItem(string name);  
//this operation adds the item to the cart.  
  
removeItem(string name);  
//removes the item from the shopping cart  
//If there are multiple items with the same name,  
remove the first occurrence. If no item is found,  
please print a message to the user indicating so.  
  
listItems();  
//display the items inside the cart.  
  
int getNumOfItems ()  
//Function to return the number of items in the  
cart.  
  
clearCart ()  
//empty the cart  
  
saveCart()  
//save the content of the cart to a file in the  
hard drive.
```

Please overload the insertion operator so it can display the content of the cart.

Feel free to add any additional helper functions or functionality that you think is needed.

In addition to the name of the item, please include the price of the item and provide a function named total that computes the costs of all items in the shopping cart.

Please pay attention to:

- The initial size of your cart should be 0.
- The cart should grow or shrink in size as the user adds items to the cart or remove items from the cart
- Please implement proper memory management to ensure there are no memory leaks.
- Please handle all possible errors appropriately, especially for operations that could fail (eg. Failed memory allocation, removing an item that does not exist, etc..)
- Please write a driver program to test the functionality of your shopping cart program.

Submission Guidelines:

1. All programming must be done in C++
2. Please turn assignment in Canvas before the due date.
3. Please provide Separate Compilation.
4. Please submit one-page report discussing your implementation approach. You can list some of the advantages and disadvantages of your approach. Analyze the time complexity of your program.
5. Observe the format of the output.
6. Observe good programming style and practices.
7. Please provide the .sln file for your program.
8. Write a driver program to test your code
9. Include screenshot of the output of your program.

===== End of HW # 3 =====