Neil Rolf

CSC300

Program 2 – Design (updated)

**Class Declarations**

Class GroceryStack

*public variables*

* GroceryStack() – default constructor, sets top of stack pointer to null
* ~GroceryStack () – deconstructor
* void setSelection(int) – sets userSelection field
* void push(string, int, double) – inserts a list node at the top of the stack, accepts name, quantity, price.
* void pop(&string, &int, &double) – removes a list node from the top of the stack, accepts pointers to variables for popped elements to be stored
* double total() const; - wrapper for recursive totalValue function
* void view() – displays contents of stack (more for debug than displaying total purchase values)

*private variables*

* struct StackNode – holds data values and pointer to next node
  + string – item name
  + int – item quantity
  + double – item cost
  + StackNode \*- pointer to next node
* StackNode \*head – points to the first node address
* double totalValue(StackNode \*) const; - recursive function for finding total value of purchases
* int userSelection; - stores user’s selection of number of items to view

**Main**

int(main){

//read data from file

ingestData(); //reads file data into class structure

if(file read unsuccessful){

exit}

while (fileobject not end of file)

getline(elements to temporary structure)

Stack.push(push from temporary structure to stack)

//user menu

display menu to user and prompt for user selection

switch(user selected value)

case 1: GroceryList.peek(user selection);

prompt user for number of items to print

calls peek function that prints information

case 2: GroceryList.total();

calls recursive total function that returns the total

prints total

case 3: exit();

uses sentinel value to exit program

} while(exit case not selected)

}

