IPC144SCP 2191.

Grocery Store Inventory System

Final Project (V1.2 – fixed const struct problem and added retargeting instructions)

In a grocery store, in order to be able to always have the proper number of items available on shelves, an inventory system is needed to keep track of items available in the inventory and make sure the quantity of the items does not fall below a specific count.

Your job for this project is to prepare an application that manages the inventory of items needed for a grocery store. The application should be able to keep track of the following information about an item:

1- The SKU Number
2- The name (maximum of 20 chars)
3- Quantity (On hand quantity currently available in the inventory)
4- Minimum Quantity (if the quantity of the item falls less than or equal to this value, a warning should be generated)
5- Price of the item
6- Is the item Taxed

This application must be able to do the following tasks:

1- Print a detailed list of all the items in the inventory
2- Search and display an item by its SKU number
3- Search and display an item by its name
4- Checkout an item to be delivered to the shelf for sale
5- Add to stock items that are recently purchased for inventory (add to their quantity)
6- Add a new item to the inventory or update an already existing item
7- Delete an item (optional)

FINAL PROJECT (THE MENU SYSTEM)
FINAL ASSEMBLY:

After implementation of the 8 milestones you can use and assemble all your functions in your milestone 2 file (grapp.c) to create a complete Grocery Inventory System.

However, since some of you may have not successfully completed some of the milestones, I will provide my solution of the 8 milestones (except ms2: grapp.c and grapp.h) to you to complete the Final project (The Menu System).

Therefore, to do the final project you only need to do your work in your milestone 2 files (grapp.c and grapp.h); these are the only two files that you are submitting to me.

My implementation of the 8 milestones is exactly like yours except the return value of these two functions:
Your updateItem() and addItem() functions return void:

```
void updateItem(struct Item* itemptr);
void addItem(struct Item item[], int *NoOfRecs, int sku);
```

My functions return int (a true or false value based on the success of the functions):

```
int updateItem(struct Item* itemptr);
int addItem(struct Item item[], int *NoOfRecs, int sku);
```

> *Check the itemSRA.h header file and compare it to yours.*
> *By the returned true or false values, you can check and see if the user aborted the action or not.*

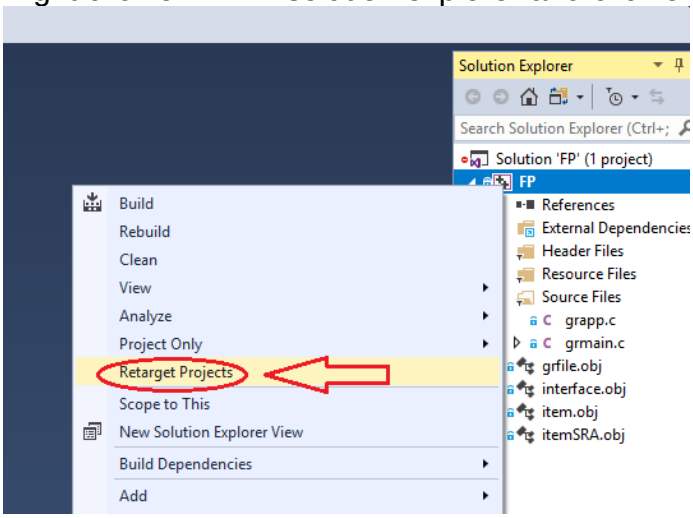
To start working on the project download the final project start-up files from here:

https://github.com/Seneca-144100/IPC144SCP-Notes/tree/master/Project/FinalProject/FP

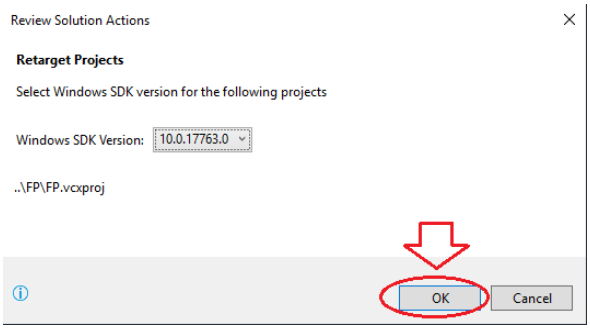
You will notice that instead of Item.c, interface.c, grfile.c and itemSRA.c files there are Item.obj, interface.obj, grfile.obj and itemSRA.obj in the project directory.
By doing this I am providing all the functions of the project without giving you their source code.

Add your grapp.c and grapp.h to the directory of the project and then retarget your visual studio as follows to match the version of the project on github to yours:

1- Open visual studio by opening FP.vcxproj
2- Right click on FP in solution explorer and click on Retarget Project:



3- Click on OK.



4- Click on File/Save All and then close your visual studio
5- Reopen FP.vcxproj.
6- Now the project is ready to start.

To ease the submission of the final project I have separated the submission of each menu item so you don't have to run a long and frustrating test program.

By doing this you will develop the task of the first option of the menu and submit it and then start working on the second one and so on…

Each submission has its own mark and they will all add up to 115%; 100 marks for menu items 1,2,3,4,5 and 7 and 15 bonus mars for menu item 6.

Here are marks for each option:

```
1- List all items                16%
2- Search by SKU                 16%
3- Checkout an item              16%
4- Stock an item                 16%
5- Add new item or update item   20%
6- Delete item                   15%  (bonus mark)
7- Search by name                16%
                      Total     115%
```

Each menu item is submitted on matrix as follows:
Upload your grapp.c and grapp.h to matrix after each stage and submit your work with this command:

~fardad.soleimanloo/submit 144_fp/X <ENTER>

replace the 'X' with the menu item number. For example, to submit the "list all items" section (menu item 1) you will submit:

~fardad.soleimanloo/submit 144_fp/1 <ENTER>

To see the due date of the project add -due to the end of the submission line:

~fardad.soleimanloo/submit 144_fp/1 -due <ENTER>

You can also apply these options if needed:

```
  "-skip_spaces":
       Do the submission regardless of incorrect horizontal spacing.
       This option may attract penalty.
  "-skip_blank_lines":
       Do the submission regardless of incorrect vertical spacing.
       This option may attract penalty.
```

TESTER PROGRAMS:

There are 2 tester files containing main() functions:
grmain.c to test menu items 1,2 and 7
grmain2.c to test menu items: 3,4,5 and 9;

grmain.c: (runs the program once and shows the content of data file)

```c
#define _CRT_SECURE_NO_WARNINGS
#include "grapp.h"
#include <stdio.h>
void showFile(void) {
  FILE* fptr = fopen(DATAFILE, "r");
  int ch;
  printf("---- Data content after execution ----------------\n");
  while ((ch = fgetc(fptr)) != EOF) putchar(ch);
  printf("--------------------------------------------------\n");
  fclose(fptr);
}
int main(void) {
  GroceryInventorySystem();
  showFile();
  return 0;
}
```

grmain2.c: (exactly like the above but runs the test twice to check the data saving)

```c
#define _CRT_SECURE_NO_WARNINGS
#include "grapp.h"
#include <stdio.h>
void showFile(void) {
  FILE* fptr = fopen(DATAFILE, "r");
  int ch;
  printf("---- Data content after execution ----------------\n");
  while ((ch = fgetc(fptr)) != EOF) putchar(ch);
  printf("--------------------------------------------------\n");
  fclose(fptr);
}
int main(void) {
  GroceryInventorySystem();
  showFile();
  GroceryInventorySystem();
  showFile();
  return 0;
}
```

YOUR TASK FOR FINAL PROJECT

Your task for the final project is to modify your `void GroceryInventorySystem (void)` to manage an array of Items, by loading them from a file and then save them back in the file and any change was made to them.

To accomplish this, modify your grapp module as follows:

In grapp.h define `DATAFILE` to `"grdata.txt"`

In grapp.c:

Include all five header files.

Compile and make sure your program should work like milestone 2 and then display the data file.

Then modify **`GroceryInventorySystem`** function as follows:

First create the following mandatory array and variables:

- An array of "`MAX_ITEM_NO`"s Item structures (let's call it items)
- An integer to hold the number of items read from file, initialized to zero. (let's call it noOfItems)
- An integer flag to hold a true/false value to indicate if the data in the Items array has been changed or not. Initialize it to false. (let's call it changed)

Suggestions:

You will need to create more variables to work with the index of the Item array and temporary variables to work with other aspects of an Item and program logic. Also, for any repetitive or complex tasks you are encouraged to code new functions in grapp.c and use them.
If you have trouble using the functions created in the 8 milestones, check the codes in the tester files in each milestone; they have the best example of how the functions are to be called and used.

Before you begin implementing the menu options do the following two tasks to take care of the loading and the saving of the items array from and to a file:

- Right after the welcome message using loadItems() function read all the information kept in the `DATAFILE` into the items array and set noOfItems to the number of items read from the file. If this was not successful print the following error message and terminate the **`GroceryInventorySystem()`** function. `"%s data file not found!\n"` (%s is for the `DATAFILE` value.)

Programming hint: Remember, it is advised not to have multiple return statements in a function.

- At the end, after the user selected to exit the program, check the changed flag to see if any data is changed.

If the changed flag is true, then ask the user if she wants to save the changes using this message:
`"Save changed to the database? (Y/N): "`
If the user responds yes, then save the items using the saveItems() function; passing the items array, `DATAFILE` and the noOfItems.
If saveItems() fails, print:
`"Failed to save changes!\n"`,
otherwise print
`"Changes saved!\n".`

MENU OPTIONS IMPLEMENTATIONS:

Replace your "under construction" messages in grapp.c with following tasks:

1- If option one is selected (List all items – 16 marks):
Display all the Items in the items array using the ListItems() function and then pause().

TO SUBMIT MENU OPTION 1:
Upload your grapp.c and grapp.h to matrix and submit your work with this command (use the test data shown at the end of this document):
`~fardad.soleimanloo/submit 144_fp/1 <ENTER>`

2- If option two is selected (Search by SKU – 16 marks)
Print: `"==SKU search==\n"`
Get a valid SKU from the user using this message:
`"Please enter the SKU of the product: "`
Then using the locateItem() function check if the SKU is found in the items array and return the index of the found element.
If the index is found, display the items element at the found index, using the displayItem() function otherwise print the following message:
`"Item not found!\n"`
Then pause using the pause() function.

TO SUBMIT MENU OPTION 2:
Upload your grapp.c and grapp.h to matrix and submit your work with this command (use the test data shown at the end of this document):
`~fardad.soleimanloo/submit 144_fp/2 <ENTER>`

3- If option three is selected (Checkout an item – 16 marks)
Print: `"==Checkout Item==\n"`
Get a valid SKU from the user using this message:
`"Please enter the SKU of the product: "`
Then using the locateItem() function check if the SKU is found in the items array and return the index of the found element.

If the index is found, display the items element at the found index, using the displayItem() function and then print:
`"Enter number of items to checkout: "`
Then receive a valid integer for the number of items to be checked out. This number must be between 1 and the quantity of the found item.
Then reduce the quantity of the found item by the number user entered, and then print:
`"%d items removed from inventory!\n".` (%d is for the number the user entered.)
Set the changed flag to true, since you just modified the items array.
If the SKU is not found print:
`"Item not found!\n"`
Then pause using the pause() function.

TO SUBMIT MENU OPTION 3:

Upload your grapp.c and grapp.h to matrix and submit your work with this command (use the test data shown at the end of this document):
`~fardad.soleimanloo/submit 144_fp/3 <ENTER>`

4- If option four is selected (Stock an item – 16 marks)
Print: `"==Stock Item==\n"`
Get a valid SKU from the user using this message:
`"Please enter the SKU of the product: "`
Then using the locateItem() function check if the SKU is found in the items array and return the index of the found element.
If the index is found, display the items element at the found index, using the displayItem() function and then print:
`"Enter number of items to stock: "`
Then receive a valid integer for the number of items to be checked out. This number must be between 1 and (999 - quantity of the found item).
Then increase the quantity of found item by the number user entered and then print:
`"%d items added to stock!\n".` (%d is for the number the user entered).
Set the changed flag to true, since you just modified the items array.
If the SKU is not found print:
`"Item not found!\n"`
Then pause using the pause() function.

TO SUBMIT MENU OPTION 4:

Upload your grapp.c and grapp.h to matrix and submit your work with this command (use the test data shown at the end of this document):
`~fardad.soleimanloo/submit 144_fp/4 <ENTER>`

5- If option five is selected (Add new item or update – 20 marks)
Print: `"==Add or Update Item==\n"`
Get a valid SKU from the user using this message:
`"Please enter the SKU of the product: "`

Then using the locateItem() function check if the SKU is found in the items array and return the index of the found element.
If SKU is found print: `"Item to edit:\n"` ,
then display it using the displayItem() function and then update the found item using updateItem() function. If updateItem() returns true, set the flag changed to true.
If the SKU is not found print:
`"Item not found, add a new one? (Y/N): "`
Using the yes() function receive the user's response. If the user's response is "No" then print `"Aborted!\n",` otherwise call the addItem() function and pass the items array, the address of noOfItems and the SKU user entered. If the function returns true set the changed flag to 1, otherwise print `"Aborted!\n".`
Then pause using the pause() function.

TO SUBMIT MENU OPTION 5:
Upload your grapp.c and grapp.h to matrix and submit your work with this command (use the test data shown at the end of this document):
`~fardad.soleimanloo/submit 144_fp/5 <ENTER>`

6- If option six is selected (Delete item – 15 bonus marks)
Print: `"==Delete Item==\n"`
Get a valid SKU from the user using this message:
`"Please enter the SKU of the product: "`
Then using the locateItem() function check if the SKU is found in the items array and return the index of the found element.
If it is found, display it using the displayItem() function and print:
`"The above Item will be deleted, are you sure? (Y/N): "`
Using the yes() function receive the user's response. If the user's response is "No" then print `"Aborted!\n",` otherwise shift all the items in the array to the left, starting from the last element back to and over the found item in the array. Then reduce the noOfItems by one); this will delete and overwrite the item.
Set the changed flag to 1; since you just modified the array.
Then pause using the pause() function.

TO SUBMIT MENU OPTION 6:
Upload your grapp.c and grapp.h to matrix and submit your work with this command (use the test data shown at the end of this document):
`~fardad.soleimanloo/submit 144_fp/6 <ENTER>`

7- If option seven is selected (Search by name – 16 marks)
Print: `"==Name search==\n"`
Get the item name in a c-string from user using this message:
`"Please enter product name: "`

Then loop through all the items and compare the names of the items to the name you just received from the user. If not found print:
`"Not found!\n"`
If found display the item using displayItem() function.
Then pause using the pause() function.

TO SUBMIT MENU OPTION 7:

Upload your grapp.c and grapp.h to matrix and submit your work with this command (use the test data shown at the end of this document):
`~fardad.soleimanloo/submit 144_fp/7 <ENTER>`

TESTING DATA:

Values in red are data entries:

Menu option one (List all Items)

```
---=== Grocery Inventory System ===---


1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 1
Row |SKU| Name               | Price  |Taxed| Qty | Min |   Total    |Atn
----+---+--------------------+--------+-----+-----+-----+------------|---
1   |275|Royal Apples        |    4.40|   No|  10 |   2 |       44.00|
2   |386|Watermelon          |    5.99|   No|  20 |   4 |      119.80|
3   |240|Blueberries         |    3.99|   No|  47 |   5 |      187.53|
4   |355|Chicken Alfredo     |    4.49|  Yes|  20 |   5 |      101.47|
5   |846|Veal Parmigiana     |    5.49|  Yes|   3 |   5 |       18.61|***
6   |359|Beefsteak Pie       |    5.29|  Yes|  40 |   5 |      239.11|
-------------------------------------------------------+---------------
                                          Grand Total: |      710.52
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,47,5,3.99,0,Blueberries
```

```
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
--------------------------------------------------
```

Menu option two (Search by SKU)

```
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 2
==SKU search==
Please enter the SKU of the product: abc
Invalid integer, please try again: 1000
Invalid value, 100 < value < 999: 111
Item not found!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 2
==SKU search==
Please enter the SKU of the product: 240
        SKU: 240
       Name: Blueberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,47,5,3.99,0,Blueberries
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
```

```
359,40,5,5.29,1,Beefsteak Pie
-----------------------------------------------
```

Menu option three (Checkout an item)

Note that the program runs 2 times by grmain2.c

```
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 3
==Checkout Item==
Please enter the SKU of the product: 240
        SKU: 240
       Name: Blueberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
Enter number of items to checkout: 50
Invalid value, 1 < value < 47: 45
45 items removed from inventory!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 1
Row |SKU| Name               | Price  |Taxed| Qty | Min |   Total    |Atn
----+---+--------------------+--------+-----+-----+-----+------------|---
1   |275|Royal Apples        |    4.40|   No|  10 |   2 |       44.00|
2   |386|Watermelon          |    5.99|   No|  20 |   4 |      119.80|
3   |240|Blueberries         |    3.99|   No|   2 |   5 |        7.98|***
4   |355|Chicken Alfredo     |    4.49|  Yes|  20 |   5 |      101.47|
5   |846|Veal Parmigiana     |    5.49|  Yes|   3 |   5 |       18.61|***
6   |359|Beefsteak Pie       |    5.29|  Yes|  40 |   5 |      239.11|
-------------------------------------------------------+---------------
                                          Grand Total: |      530.97
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
```

```
Save changed to the database? (Y/N): n
Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,47,5,3.99,0,Blueberries
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 3
==Checkout Item==
Please enter the SKU of the product: 240
        SKU: 240
       Name: Blueberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
Enter number of items to checkout: 45
45 items removed from inventory!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Save changed to the database? (Y/N): y
Changes saved!

Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,2,5,3.99,0,Blueberries
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
```

Menu option four (Stock an item)

Note that the program runs 2 times by grmain2.c

```
---=== Grocery Inventory System ===---
```

```
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 4
==Stock Item==
Please enter the SKU of the product: 240
        SKU: 240
       Name: Blueberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
Enter number of items to stock: 960
Invalid value, 1 < value < 952: 2
2 items added to stock!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 1
Row |SKU| Name               | Price  |Taxed| Qty | Min |   Total    |Atn
----+---+--------------------+--------+-----+-----+-----+------------|---
1   |275|Royal Apples        |    4.40|   No|  10 |   2 |       44.00|
2   |386|Watermelon          |    5.99|   No|  20 |   4 |      119.80|
3   |240|Blueberries         |    3.99|   No|  49 |   5 |      195.51|
4   |355|Chicken Alfredo     |    4.49|  Yes|  20 |   5 |      101.47|
5   |846|Veal Parmigiana     |    5.49|  Yes|   3 |   5 |       18.61|***
6   |359|Beefsteak Pie       |    5.29|  Yes|  40 |   5 |      239.11|
-------------------------------------------------------+----------------
                                          Grand Total: |      718.50
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Save changed to the database? (Y/N): n
Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,47,5,3.99,0,Blueberries
```

```
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 4
==Stock Item==
Please enter the SKU of the product: 240
        SKU: 240
       Name: Blueberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
Enter number of items to stock: 2
2 items added to stock!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Save changed to the database? (Y/N): y
Changes saved!

Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,49,5,3.99,0,Blueberries
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
```

Menu option five (Add new item or update item)

Note that the program runs 2 times by grmain2.c

```
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
```

```
7- Search by name
0- Exit program
> 5
==Add or Update Item==
Please enter the SKU of the product: 111
Item not found, add a new one? (Y/N): y
        SKU: 111
       Name: a
      Price: 1
   Quantity: 1
Minimum Qty: 1
   Is Taxed: y
Add Item? (Y)es/(N)o: y
Item Added!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 5
==Add or Update Item==
Please enter the SKU of the product: 222
Item not found, add a new one? (Y/N): y
        SKU: 222
       Name: a
      Price: 1
   Quantity: 1
Minimum Qty: 1
   Is Taxed: y
Add Item? (Y)es/(N)o: n
Aborted!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 5
==Add or Update Item==
Please enter the SKU of the product: 222
Item not found, add a new one? (Y/N): n
Aborted!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
```

```
7- Search by name
0- Exit program
> 5
==Add or Update Item==
Please enter the SKU of the product: 240
Item to edit:
        SKU: 240
       Name: Blueberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
Enter new data:
        SKU: 240
       Name: Blackberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: n
Overwrite old data? (Y)es,(N)o: y
Item updated!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 5
==Add or Update Item==
Please enter the SKU of the product: 240
Item to edit:
        SKU: 240
       Name: Blackberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
Enter new data:
        SKU: 240
       Name: a
      Price: 1
   Quantity: 1
Minimum Qty: 1
   Is Taxed: y
Overwrite old data? (Y)es,(N)o: n
Overwrite Aborted!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
```

```
7- Search by name
0- Exit program
> 1
Row |SKU| Name               | Price  |Taxed| Qty | Min |   Total    |Atn
----+---+--------------------+--------+-----+-----+-----+------------|---
1   |275|Royal Apples        |    4.40|   No|  10 |   2 |       44.00|
2   |386|Watermelon          |    5.99|   No|  20 |   4 |      119.80|
3   |240|Blackberries        |    3.99|   No|  47 |   5 |      187.53|
4   |355|Chicken Alfredo     |    4.49|  Yes|  20 |   5 |      101.47|
5   |846|Veal Parmigiana     |    5.49|  Yes|   3 |   5 |       18.61|***
6   |359|Beefsteak Pie       |    5.29|  Yes|  40 |   5 |      239.11|
7   |111|a                   |    1.00|  Yes|   1 |   1 |        1.13|***
-------------------------------------------------------+---------------
                                          Grand Total: |      711.65
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Save changed to the database? (Y/N): n
Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,47,5,3.99,0,Blueberries
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 5
==Add or Update Item==
Please enter the SKU of the product: 111
Item not found, add a new one? (Y/N): y
        SKU: 111
       Name: a
      Price: 1
   Quantity: 1
Minimum Qty: 1
   Is Taxed: y
Add Item? (Y)es/(N)o: y
Item Added!
```

```
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Save changed to the database? (Y/N): y
Changes saved!

Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,47,5,3.99,0,Blueberries
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
111,1,1,1.00,1,a
-------------------------------------------------
```

Menu option six (Delete item) – optional, do this for bonus marks

Note that the program runs 2 times by grmain2.c

```
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 6
==Delete Item==
Please enter the SKU of the product: 240
        SKU: 240
       Name: Blueberries
      Price: 3.99
   Quantity: 47
Minimum Qty: 5
   Is Taxed: No
The above Item will be deleted, are you sure? (Y/N): y
Item Deleted!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 6
```

```
==Delete Item==
Please enter the SKU of the product: 355
        SKU: 355
       Name: Chicken Alfredo
      Price: 4.49
   Quantity: 20
Minimum Qty: 5
   Is Taxed: Yes
The above Item will be deleted, are you sure? (Y/N): n
Aborted!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 1
Row |SKU| Name               | Price  |Taxed| Qty | Min |   Total    |Atn
----+---+--------------------+--------+-----+-----+-----+------------|---
1   |275|Royal Apples        |    4.40|   No|  10 |   2 |       44.00|
2   |386|Watermelon          |    5.99|   No|  20 |   4 |      119.80|
3   |355|Chicken Alfredo     |    4.49|  Yes|  20 |   5 |      101.47|
4   |846|Veal Parmigiana     |    5.49|  Yes|   3 |   5 |       18.61|***
5   |359|Beefsteak Pie       |    5.29|  Yes|  40 |   5 |      239.11|
-------------------------------------------------------------+----------------
                                                Grand Total: |      522.99
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Save changed to the database? (Y/N): y
Changes saved!

Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
```

```
6- Delete item
7- Search by name
0- Exit program
> 6
==Delete Item==
Please enter the SKU of the product: 355
        SKU: 355
       Name: Chicken Alfredo
      Price: 4.49
   Quantity: 20
Minimum Qty: 5
   Is Taxed: Yes
The above Item will be deleted, are you sure? (Y/N): y
Item Deleted!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 1
Row |SKU| Name               | Price  |Taxed| Qty | Min |   Total    |Atn
----+---+--------------------+--------+-----+-----+-----+------------|---
1   |275|Royal Apples        |    4.40|   No|  10 |   2 |       44.00|
2   |386|Watermelon          |    5.99|   No|  20 |   4 |      119.80|
3   |846|Veal Parmigiana     |    5.49|  Yes|   3 |   5 |       18.61|***
4   |359|Beefsteak Pie       |    5.29|  Yes|  40 |   5 |      239.11|
-------------------------------------------------------+----------------
                                          Grand Total: |      421.52
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Save changed to the database? (Y/N): n
Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
```

Menu option seven (Search by name)

```
---=== Grocery Inventory System ===---

1- List all items
2- Search by SKU
3- Checkout an item
```

```
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 7
==Name Search==
Please enter product name: hoohoo
Not found!
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 7
==Name Search==
Please enter product name: Watermelon
        SKU: 386
       Name: Watermelon
      Price: 5.99
   Quantity: 20
Minimum Qty: 4
   Is Taxed: No
Press <ENTER> to continue... <ENTER>
1- List all items
2- Search by SKU
3- Checkout an item
4- Stock an item
5- Add new item or update item
6- Delete item
7- Search by name
0- Exit program
> 0
Exit the program? (Y)es/(N)o: y
Goodbye!
---- Data content after execution ----------------
275,10,2,4.40,0,Royal Apples
386,20,4,5.99,0,Watermelon
240,47,5,3.99,0,Blueberries
355,20,5,4.49,1,Chicken Alfredo
846,3,5,5.49,1,Veal Parmigiana
359,40,5,5.29,1,Beefsteak Pie
-------------------------------------------------
```