Homework 12

CS-102

Spring 2022

Homework 12

- Read **all** of Homework 12A before proceeding with its solution.
- Homework 12 is broken up into two parts where Homework 12A will be due next week, while Homework 12B will be due in two weeks.
- These two homeworks, when put together in one program, will comprise a menu driven application which is stored in a Random file management system.
- In addition, you will be using binary files of **struct** in order to hold and access the data.
- Chapters 11 and 12 will supply for you the background knowledge for you to be able to create this system.

Background Information

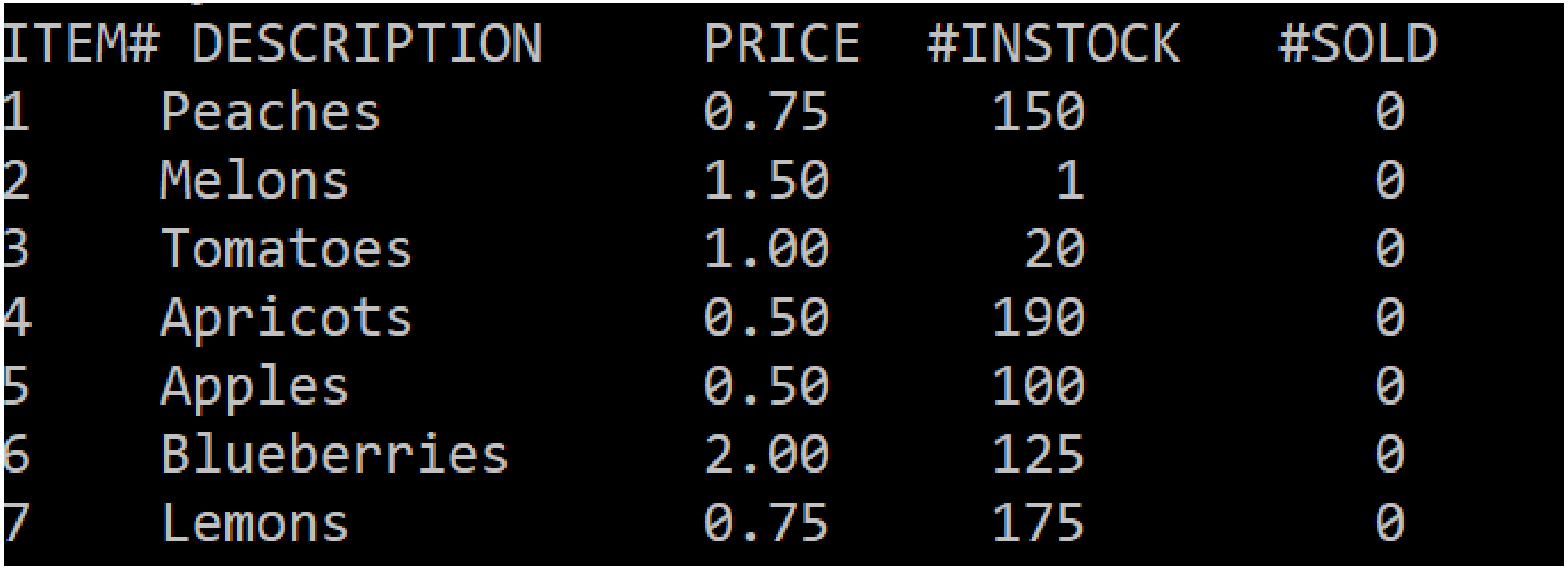
- A large family farm is producing locally grown fruit and vegetables.
- Each week they deliver orders from the produce that they grow to local dropoff points where we pick up our weekly, freshly grown, produce.
- They are interested in creating an ordering mechanism by which people can go online and select what they want to go into their weekly baskets.
- Your job is to create software that will allow them to take custom orders on-line that they will fill in boxes with your names on them.
- The data, you will be using, is simply a fruit simulation of the data that they might actually use. Nonetheless, the principles will be the same.

Homework 12A

- Create a **struct** called **Item** to store information about item of produce that is grown by local farmers. The elements of the struct are:
  - Description (use an array of char where SIZE = 20)
  - Price (use a double)
  - quantity in stock (use an int)
  - quantity sold (use an int)
- **Create Random File made up of records of Item** to represent all the various fruits and vegetables that the local farms will be producing.
  - Use 10 as the maximum number of items in that will eventually be in inventory.
  - Use 7 as the number of items currently in the inventory.
  - Use 20 as the maximum size of the array of char that will hold the name of the item.

Homework 12A

- Here is the initial inventory of your store that you will be entering into your file management system.



```
ITEM# DESCRIPTION     PRICE  #INSTOCK   #SOLD
1    Peaches          0.75     150         0
2    Melons           1.50       1         0
3    Tomatoes         1.00      20         0
4    Apricots         0.50     190         0
5    Apples           0.50     100         0
6    Blueberries      2.00     125         0
7    Lemons           0.75     175         0
```

Homework 12A

- **These functions are required for full credit.**
  - createEmptyInventory()
  - editInventory()
  - displayInventory()
  - makeAnOrder() // For Hwrk12A, this can be a stub
  - createOrderReport() // For Hwrk12A, this can be a stub
  - createRevenueReport() // For Hwrk12A, this can be a stub
  - saveInventory() // For Hwrk12A, this can be a stub

Homework 12 Requirements

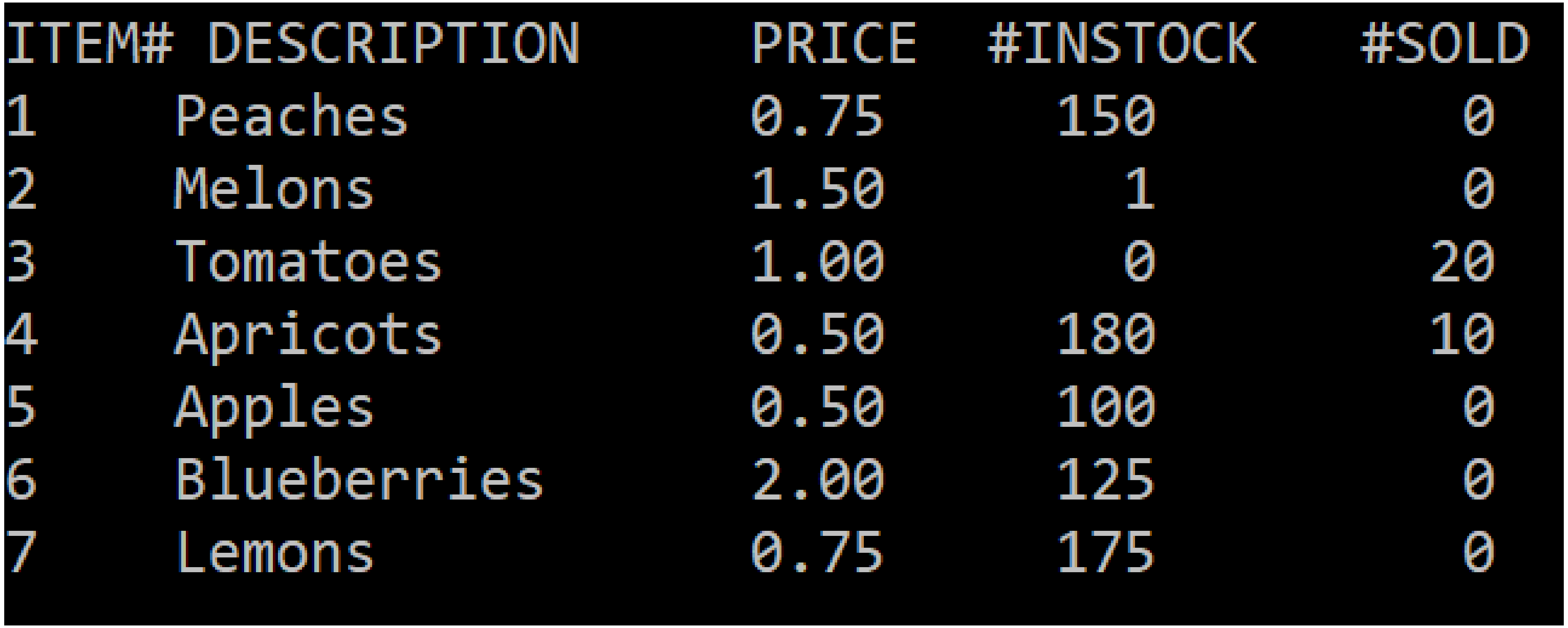
- Displays the list of items available at the local farm. (In Homework 12A & 12B)
- Orders are processed by selecting an item number. Note that item 1 is stored in element 0 of the array. (Your program must adjust accordingly!
- For each order: (Note this requirement will appear in Homework 12B.)
    - Make sure a valid item number is selected
    - Make sure item is in stock
    - Ask for quantity (use a loop to validate that it is positive)
    - Make sure there is enough stock for this order
    - Show the total amount of purchase
    - Update the stock and sold
    - This continues until -1 is entered to close the order.
- Generate revenue report (see sample output) (Homework 12B)
- Generate re-order report (any products with less than 5 in stock) (Homework 12B)
- Save the inventory.dat file at the end of each day, so you have a backup record. (Homework 12B)

Homework 12A createEmptyInventory() Requirements

- This is done once, to initialize the Random File.
- When createEmptyInventory() is run, it should warn the user that running this function will destroy all data currently stored in the file.
  - It should then pause to allow the user to **not** use this option.
- If the user chooses to run it, MAX_RECORDS = 10 are created in the file: **inventory.dat** .
  - These records will contain zero for all the numeric fields of **struct item**, and will contain null data **“”** for the field **description**.

Homework 12A
displayInventory()

- This function will display the entire contents of inventory.dat at any point during the day's transactions.



```
ITEM# DESCRIPTION     PRICE  #INSTOCK   #SOLD
1    Peaches          0.75     150         0
2    Melons           1.50       1         0
3    Tomatoes         1.00       0        20
4    Apricots         0.50     180        10
5    Apples           0.50     100         0
6    Blueberries      2.00     125         0
7    Lemons           0.75     175         0
```

Homework 12A
editInventory()

- This function will display all the records on the screen and then ask you which one you would like to edit.
- After selecting the record number that you would like to edit, you are asked the following questions
  - “Which Item # do you want to edit: “
  - You type in the record number that you want to edit and hit Enter.
  - It then asks you, with what values you wish to replace all the fields in the record.
    - description
    - price
    - quantityInStock
    - quantitySold
- After entering the values, it then asks you,
  - “Do you want to edit another record? “
  - To which you would answer, “y” or “n”.

Homework 12A Cont.

- Although the remaining options would appear on the menu, they only need to be stubs until you do Homework 12B which is due the following week.
- For example, the stub for “Make an Order” might look like this:

```
void makeOrder();
{
  cout << “This option will run the Make an Order function” << endl;
  cin.ignore();
  cin.get();
}
```

Tips for doing Homework 12

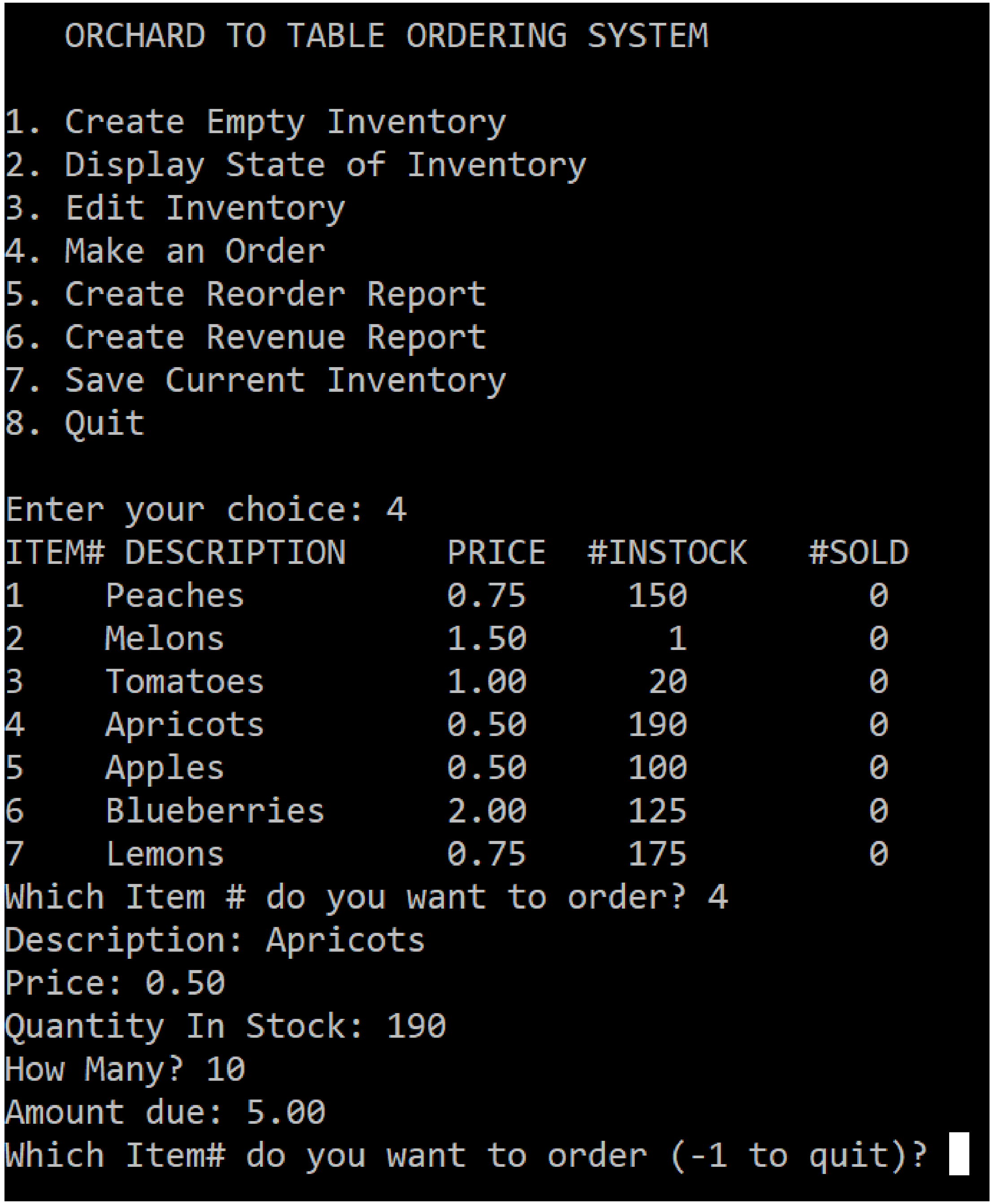
- Do one function at a time, and get it working, before entering in the code for the next function.
  - This may mean, at the beginning, that you will be working on the first function, createInventory(), and making the remaining 6 functions, just stubs.
  - In this way, the menu, itself will look like the final result that you are seeking even though, six out of seven of the functions are simply stubs.
- You will find the following programs helpful in doing assignments 12A and 12B.
  - Programs 12-20, 12-21 and 12-22 will be of great assistance for Homework 12A.
  - Program 6-10 will refresh your memory on how to set up a menu driven program.

Homework 12B

- **These functions are required for full credit.**
  - createEmptyInventory() // This has already been done for Hwrk12A
  - editInventory() // This has already been done for Hwrk12A
  - displayInventory() // This has already been done for Hwrk12A
  - makeAnOrder() // For Hwrk12B
  - createOrderReport() // For Hwrk12B
  - createRevenueReport() // For Hwrk12B
  - saveInventory() // For Hwrk12B

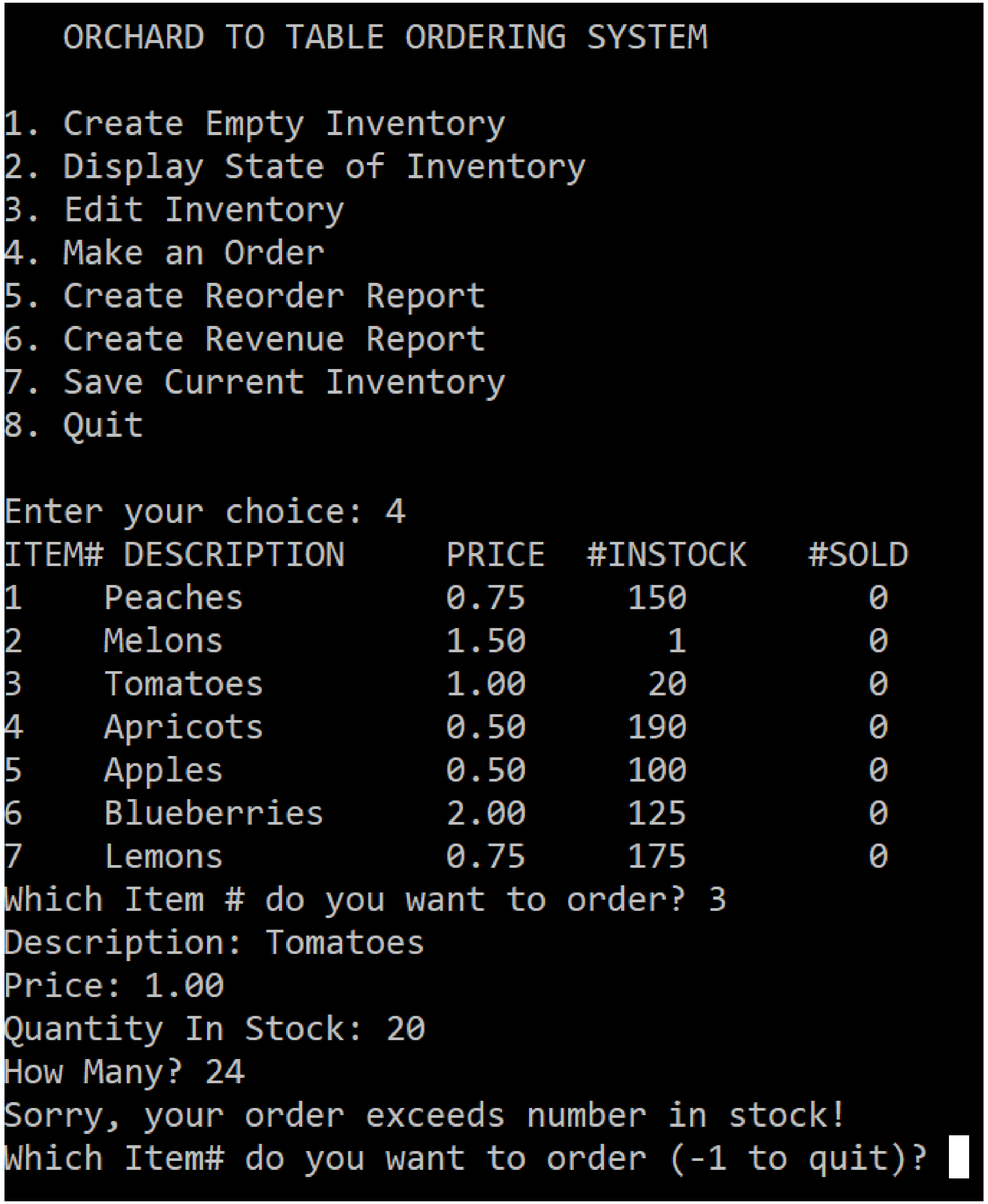
Homework 12B Requirements for makeAnOrder()

- When you select “Make an Order” on the Main Menu, you will then see displayed the produce that you can choose from.
- Here is an example of the kind of order you might make



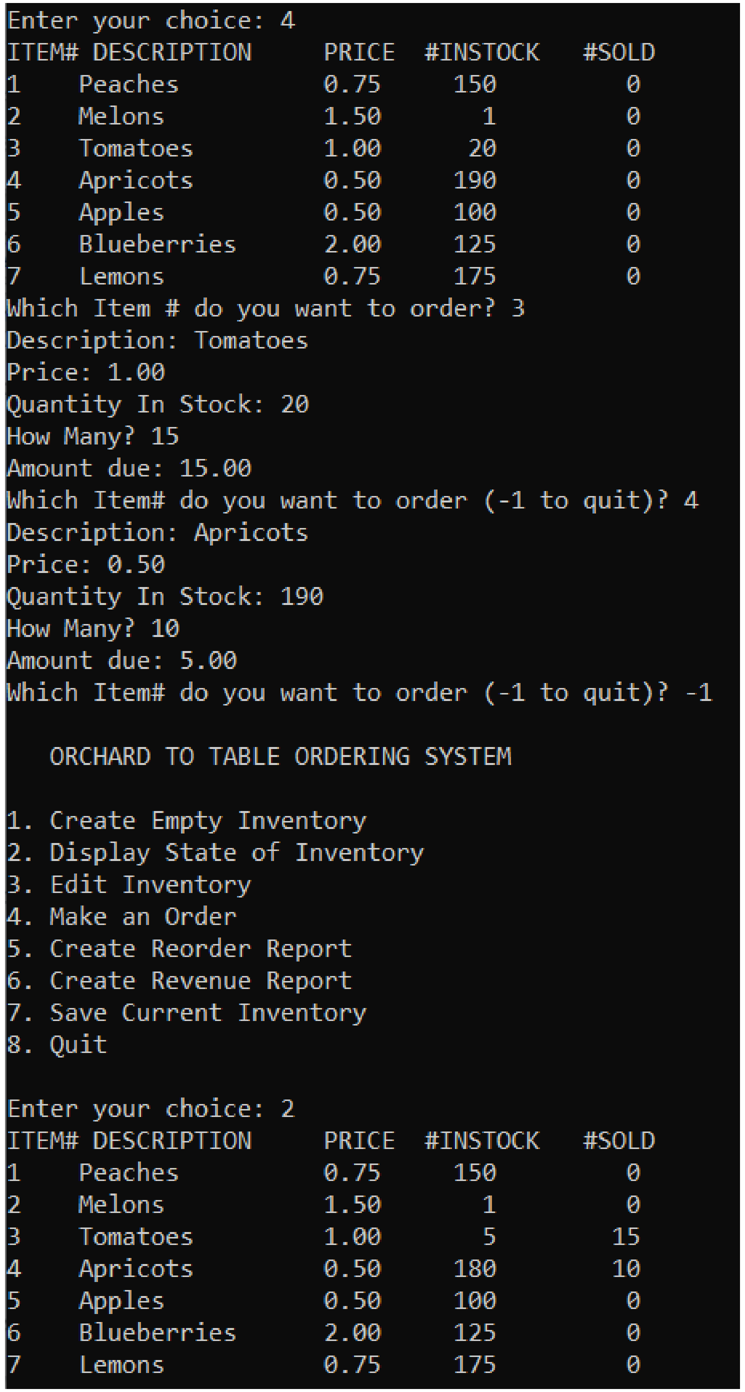
```
   ORCHARD TO TABLE ORDERING SYSTEM

1. Create Empty Inventory
2. Display State of Inventory
3. Edit Inventory
4. Make an Order
5. Create Reorder Report
6. Create Revenue Report
7. Save Current Inventory
8. Quit

Enter your choice: 4
ITEM# DESCRIPTION     PRICE  #INSTOCK   #SOLD
1    Peaches          0.75     150         0
2    Melons           1.50       1         0
3    Tomatoes         1.00      20         0
4    Apricots         0.50     190         0
5    Apples           0.50     100         0
6    Blueberries      2.00     125         0
7    Lemons           0.75     175         0
Which Item # do you want to order? 4
Description: Apricots
Price: 0.50
Quantity In Stock: 190
How Many? 10
Amount due: 5.00
Which Item# do you want to order (-1 to quit)?
```

Homework 12B Requirements for makeAnOrder()



```
   ORCHARD TO TABLE ORDERING SYSTEM

1. Create Empty Inventory
2. Display State of Inventory
3. Edit Inventory
4. Make an Order
5. Create Reorder Report
6. Create Revenue Report
7. Save Current Inventory
8. Quit

Enter your choice: 4
ITEM# DESCRIPTION     PRICE  #INSTOCK   #SOLD
1    Peaches          0.75     150         0
2    Melons           1.50       1         0
3    Tomatoes         1.00      20         0
4    Apricots         0.50     190         0
5    Apples           0.50     100         0
6    Blueberries      2.00     125         0
7    Lemons           0.75     175         0
Which Item # do you want to order? 3
Description: Tomatoes
Price: 1.00
Quantity In Stock: 20
How Many? 24
Sorry, your order exceeds number in stock!
Which Item# do you want to order (-1 to quit)?
```

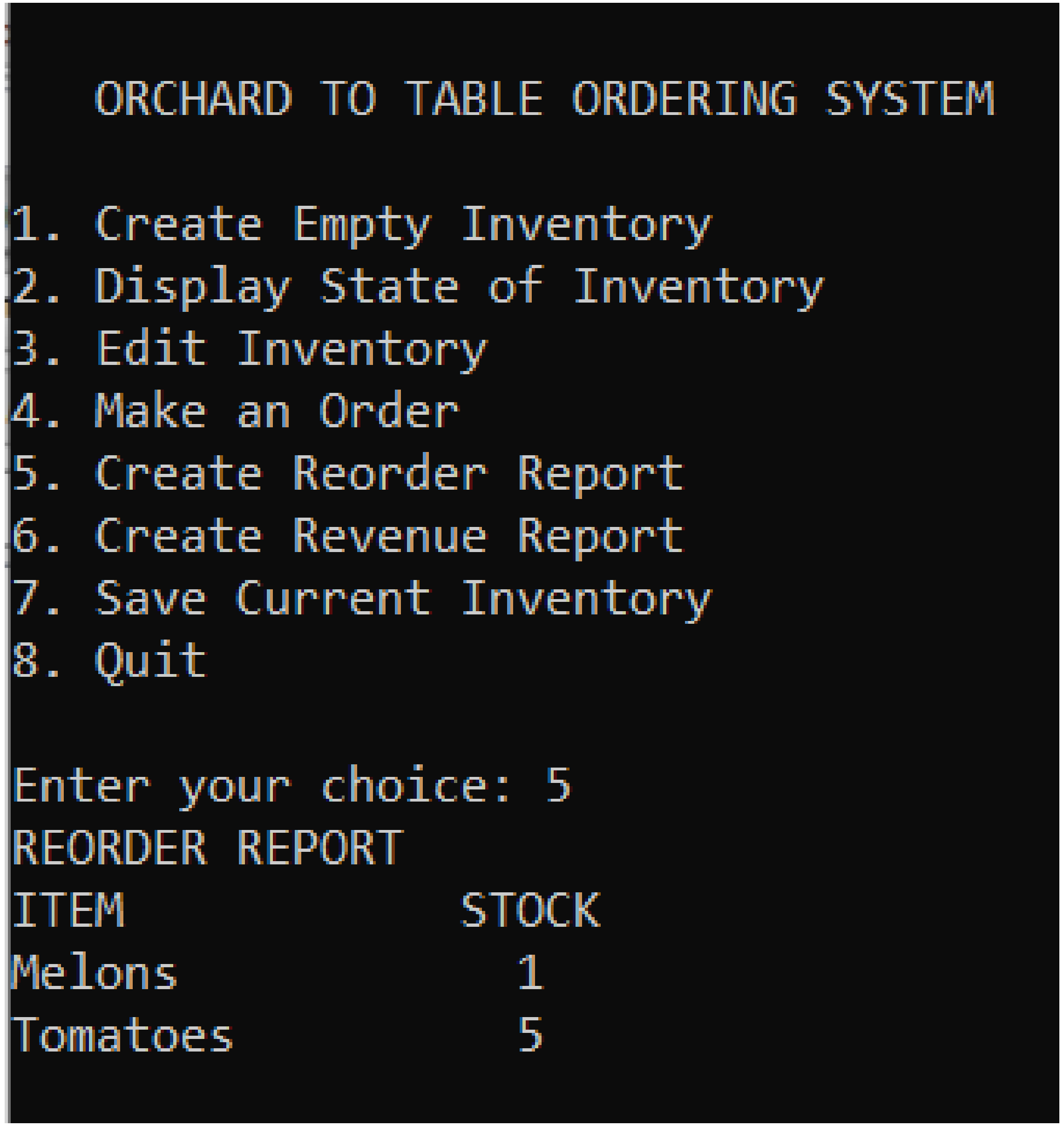
- If you order more items than are in stock, here is the kind of response you would get.



```
Enter your choice: 4
ITEM# DESCRIPTION     PRICE  #INSTOCK   #SOLD
1    Peaches          0.75     150         0
2    Melons           1.50       1         0
3    Tomatoes         1.00      20         0
4    Apricots         0.50     190         0
5    Apples           0.50     100         0
6    Blueberries      2.00     125         0
7    Lemons           0.75     175         0
Which Item # do you want to order? 3
Description: Tomatoes
Price: 1.00
Quantity In Stock: 20
How Many? 15
Amount due: 15.00
Which Item# do you want to order (-1 to quit)? 4
Description: Apricots
Price: 0.50
Quantity In Stock: 190
How Many? 10
Amount due: 5.00
Which Item# do you want to order (-1 to quit)? -1

   ORCHARD TO TABLE ORDERING SYSTEM

1. Create Empty Inventory
2. Display State of Inventory
3. Edit Inventory
4. Make an Order
5. Create Reorder Report
6. Create Revenue Report
7. Save Current Inventory
8. Quit

Enter your choice: 2
ITEM# DESCRIPTION     PRICE  #INSTOCK   #SOLD
1    Peaches          0.75     150         0
2    Melons           1.50       1         0
3    Tomatoes         1.00       5        15
4    Apricots         0.50     180        10
5    Apples           0.50     100         0
6    Blueberries      2.00     125         0
7    Lemons           0.75     175         0
```

Homework 12B Requirements for makeAnOrder()

- Now suppose you order some Tomatoes that are in stock, as well as some Apricots.
- When you Display State of Inventory again, you see the number of Tomatoes that have been sold as well as the number of Apricots.
- We could have used the Edit menu item to make these purchases, but the Make An Order menu option is more efficient.

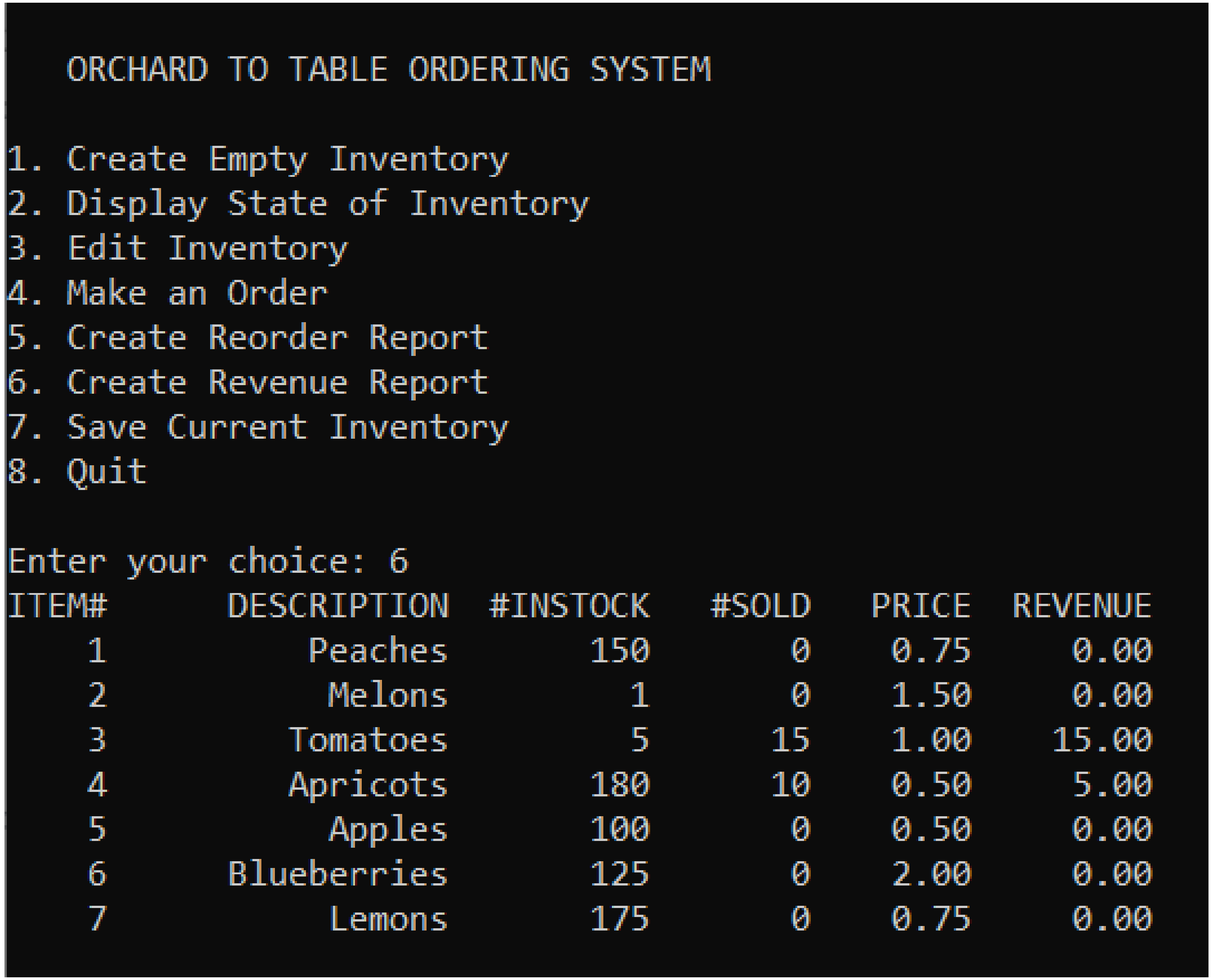


```
   ORCHARD TO TABLE ORDERING SYSTEM

1. Create Empty Inventory
2. Display State of Inventory
3. Edit Inventory
4. Make an Order
5. Create Reorder Report
6. Create Revenue Report
7. Save Current Inventory
8. Quit

Enter your choice: 5
REORDER REPORT
ITEM                STOCK
Melons                1
Tomatoes              5
```

Homework 12B Requirements for createOrderReport()

- Now we can generate a Re-order report which tells us which stock items we are low in.
- If we have 5 or fewer items in-stock, then the items should appear on the re-order report.

Homework 12B Requirements for createRevenueReport()



```
   ORCHARD TO TABLE ORDERING SYSTEM

1. Create Empty Inventory
2. Display State of Inventory
3. Edit Inventory
4. Make an Order
5. Create Reorder Report
6. Create Revenue Report
7. Save Current Inventory
8. Quit

Enter your choice: 6
ITEM#      DESCRIPTION  #INSTOCK   #SOLD    PRICE  REVENUE
    1          Peaches       150       0     0.75     0.00
    2           Melons         1       0     1.50     0.00
    3         Tomatoes         5      15     1.00    15.00
    4         Apricots       180      10     0.50     5.00
    5           Apples       100       0     0.50     0.00
    6      Blueberries       125       0     2.00     0.00
    7           Lemons       175       0     0.75     0.00
```

- We can generate a Revenue report which tells us how much we have made on each of the items that we carry.
- Notice that the number of items that we have sold have now shown up on our Inventory report as well as the Revenue report.

Homework 12B
Requirements for saveInventory()

- Finally, you will want to be able to save your Inventory File at the end of the day so that you know how much business you did on that particular day.
  - For example your inventory.dat file might be saved as inventory_2021-12-01.dat
- Also, this will back up your inventory, just in case you delete it by insisting on doing option 1 more than once.