CPSC 131- Data Structure : Project 1

Shrinivas Patil
spatil@csu.fullerton.edu

Submission: Upload on Canvas the program file with proper extension. Add your name and CWID in the beginning of the starter template. Or else, upload a link to the github repository and give me access.

Introduction

In this project you will design and implement vector and vector operations for the management system. You will be provided with a starter template with necessary classes and templates. You need to add functions that are specified in the document. Do not change the main function, I have added test cases in it.

Food Pantry Management System

CSUF wants to make Inventory management and appointment management system to maintain data related to donations and students who take advantage of this service. They are asking newly admitted students to design a system that tracks all the operations. The pantry management has asked you to add features to manage inventory and appointments. Your task is to create functions which will add functionalities to the management system. This are some functional requirements:

You are given with two template classes for inventory management system:

Item - Item class template decides the properties of a particular item. It has four properties: name, expiration, category, quantity.

Inventory - Inventory function will take care of all the operations. Inventory class will have one vector which will store objects of Item.

1) **Add New Item**: Write a function **addNewItem()** which will add a new Item in your vector. While adding a new element in the vector, check whether it already exists in the vector or not. If it exists in a vector, then display the message as "Item is already present in inventory".
2) **Increase Quantity**: Write a function **increaseQuantity**() to increase the quantity of the item by the newly provided quantity. It will take two parameters: itemname and quantity.
3) **Update Item**: Write a function **updateItem()** which will update the quantity, expiration and category. updateItem() will take four parameters: itemname, expiration, quantity and category. It should search the item from the name and update the item with given parameters. We can only update the items that are already present in inventory.
4) **Remove Item** : Write a function **removeItem()** which will remove a particular item from the vector on the name of item. The function removeItem() takes a parameter as itemname. If an item is not in an inventory and you are trying to remove it from inventory, it will throw an exception "Item not found".
5) **Total**: Write a function **Total()** to calculate the total number of items in inventory.
6) **Search Item**: Write a function **searchItem()** to search a particular item in the inventory. The function searchItem() takes a parameter itemname and based on that it will find an item. If an item is not present it will throw an exception "Item not found!!".

For appointment management you are provided with two classes:

Appointment- Appointment class template decides the properties of an appointment. It has four properties: name, ap_date, ap_time, CWID.

AppointmentSystem **-** This class is responsible for all the operations. AppointmentSystem class will have one vector and it will store objects of Appointment.

1) **Schedule:** Write a function **schedule()** to schedule an appointment. It will store an appointment in a vector. If CWID is already present in a vector, then display the message that "You have already scheduled an appointment!!!". It will store an object in a vector.
2) **Total_appointments:** Write a function **Total_appointments()** to calculate a total number appointments on a particular date and at a particular time. It will take two parameters: date and time.
3) **Remove recent:** Write a function **removeRecent()** to delete an appointment that is recently booked.

Grading Rubric:

Your grade will be comprised of following parts:

1) **addNewItem() (2 marks)**
2) **increaseQuantity(4 marks)**
3) **updateItem() (4 marks)**
4) **removeItem() (5 marks)**
5) **Total() (2 marks)**
6) **searchItem() (3 marks)**
7) **schedule() (3 marks)**
8) **Total_appointments(4 marks)**
9) **removeRecent() (3 marks)**
10) **Compilation of program (10 marks)**
11) **Successful run on all test cases(10 marks)- Partial marks will be given if few of them successfully run.**

What to Do
The starter template is provided. It has classes and main() function. Do not make any changes in the main function. Make changes in respective classes. Add the same name of a function that I mentioned above. If you make any other changes and code is giving different output. You will lose the marks. Make the submissions on time. No extensions will be provided. Late submissions will be accepted but 3% marks will be deducted.