CP1300 Assignment 2: Eatery Guide

Problem Description

Your snack cart project hit a success. Since the snack cart machines rolled out, business has been growing steady. Now the business owner decides to expand the company service to restaurant sector. You are approached by the company to develop an eatery system so that for a tourist visiting Beijing, the eatery system might be helpful to find snack carts and restaurant in the city, which snack cart has the best Jiānbĭng guŏzi, and which restaurant has the best Peking roast duck.

Your Task:

1. Based on assignment 1 extend the snack cart system to an eatery system, design an **Eatery** class. Create **SnackCart** and **Restaurant** classes, both classes are derived from Eatery.
2. The **Eatery** class has owner and multiple meal objects.
3. The **SnackCart** has location.
4. The **Restaurant** has name, address, telephone number and opening hour.
5. The eatery system shall provide the same functionality of assignment 1 **plus** add a new restaurant, delete an existing restaurant and load/save/display restaurant information.
6. Write a program **RestaurantTest.cpp** that performs unit tests on the Restaurant class to help you verify that it works correctly. The program needs to be commented to explain what each unit test is trying to verify
7. Re-write the program **CartTest.cpp** that performs unit tests on the Meal class to help you verify that it works correctly. The program needs to be commented to explain what each unit test is trying to verify.
8. Re-develop the program StreetSnacks.cpp and name it **EateryGuide.cpp**, so that it makes use of the SnackCart and Restaurant classes and the Meal class.
9. The Eatery class provides operator overloading functions for stream operators **<<** and **>>**. This function prints out owner, open hours and list of the meals in an eatery object.

General Functionality:

- The program should include all functions in the **general functionality** section of **assignment 1**.
- Your program code should use **exception** mechanism where appropriate to handle runtime error.
- The program should use **List** to hold a dynamic number of meals in an eatery.

Submission

Your code must compile and run cleanly on a Unix machine using **g++**. It must be able to read and write files using the format as seen in the file “eateries.txt”, not some other file format.

You are to submit a single zip file with all and only the following files (no sub-folders):

Meal.h	Meal class declaration
Meal.cpp	Meal member function definitions
Eatery.h	Eatery class declaration
Eatery.cpp	Eatery member function definitions
SnackCart.h	SnackCart class declaration
SnackCart.cpp	SnackCart member function definitions
Restaurant.h	Restaurant class declaration
Restaurant.cpp	Restaurant member function definitions
RestaurantTest.cpp	unit tests for the Meal class
CartTest.cpp	unit tests for the SnackCart class
EateryGuide.cpp	the eatery guide database program
makefile	a makefile to compile the programs: CartTest, RestaurantTest and EateryGuide
Eateries.txt	text file containing the data records from running the EateryGuide program

The assignment is to be submitted in LearnJCU. It is expected that your files have sufficient comments so that no other documentation is needed to understand the code.

Please note: when you upload the zip file you may get an error message. **IGNORE IT**. LearnJCU submission box performs plagiarism detection which will not work on zip files – the error message means your upload has worked but the plagiarism detection has not.