Homework 6

File I/O, Classes, Dynamic Allocation, Exception Handling

Request:

Design a program that will read in 2 text files and generate a list of cars and their repairs. The output should include the car's model first, then a list of repairs that the car has. If the car does not have any repairs, output "Pass: Car is done". Each line in the "Cars.txt" contain the car's model followed by any repairs the car needs. Match those numbers to the list of repairs shown in "RepairsKey.txt".

Ex:

Cars.txt	RepairsKey.txt
Corolla 2 4 5 6 Camry 0 XC60 1 4 RS5 3 6 ...	0 Car Fixed 1 Brake 2 Windshield 3 Tires 4 Engine 5 Doors 6 Transmission

Thus, the car model Corolla has repairs needed: Windshield, Engine, Doors, and Transmission.
The car model Camry is fixed therefore output: "Pass: Car is done"
The car model XC60 has repairs needed: Brake and Engine
and so on ...

Requirement:

- Keep the names of the text files you read in as "Cars.txt" and "RepairsKey.txt".
- Output each repair on a separate line with a counter to track how many there are. (see example output)

Structure:

You should organize your code with the following:

- 1 structure, RepairList to store an int *repairCode* and a string *repairPart*
- 1 class, CarLot to store the cars and their respective repairs.
  - Private variables: string *model*, int *repairSize*, int* *repairs*, static int *numOfCars*
  - Constructor, Destructor, Mutators and Accessors (get and set functions for necessary variables you wish to call/modify).
  - You can add more functions if it need be.
- You can either read in both text files in main or in a separate function.
  - Read in "RepairsKey.txt" to an array of RepairList
  - Read in "Cars.txt" to an array of CarLot

Notes:

- You should make *repairs* an int* since each car will have a random amount of repairs that is dependent on *repairSize*.

- You MUST use a try and catch block to check for when a car is fixed. You should try to find a car that has 0 repairs and handle what happens when it is caught. i.e. output "Pass: Car is done"

- You are allowed to use string arrays and/or vectors for this assignment.

- When you read "Cars.txt" into your code, be careful how you handle the numbers of repairs for each car. Since each car has a random number of repairs, if you just use a loop to read, it might crash your program. Thus, you could read in as a string of words, then split the string into an array or vector. After that set each element into the correct location.

- You do not need to output the repairCode, the repairMessage will suffice.

Example Output:

```
Car Model: Corolla
Repair #1: Windshield
Repair #2: Engine
Repair #3: Doors
Repair #4: Transmission

Car Model: Camry
Pass: Car is done

Car Model: XC60
Repair #1: Brake
Repair #2: Engine

Car Model: RS5
Repair #1: Tires
Repair #2: Transmission
```