1) Write a program in which linked lists are used to maintain the data structure for a car rental company. The program allows several types of transactions to be applied to the data structure in order to keep the lists up-to-date.
In the program three linked lists are maintained for cars:

- available for rent,
- rented,
- in repair.

The cars on the available list are ordered by mileage, with the car having the least miles at the front of the list. The cars on the rented list are ordered by the expected return dates, with the first car on this list having the earliest expected return date.
The program prompts the user for a transaction code (integer) as follows:
(a) add a new car to the available-for-rent list,
(b) add a returned car to the available-for-rent list,
(c) add a returned car to the repair list,
(d) transfer a car from the repair list to the available-for-rent list,
(e) rent the first available car,
(f) print all the lists,
(g) quit.

For the new car addition and the return transactions (codes 1 - 3), the program should then prompt for a plate number (character string) and a mileage (integer). For the transfer transaction (code 4), the program should then prompt for a plate number (character string). For the rent transaction (code 5), the program should then prompt for an expected return date (integer: yymmdd). For the print transaction (code 6), the program should not prompt any additional information. The program should quit when the user selects 7 in the prompter.

When a transaction is processed, a message should be printed indicating what action is taken, for example which car is transferred from which list to which list. For each return transaction, a charge is computed and printed as follows:

- a flat rate of $80.00 for up to 200 km,
- 15 cents per km for the additional (i.e. beyond 200) kilometres.

When a quit transaction is completed, the program prints the total income from all the rented cars.

Also when a quit transaction is completed, the program should store the data into a disk file, and when the program restarts, the program should read the data from the file and restore the lists. Please create an initial disk file so that when the program starts for the first time, there are data on the lists. The file should contain at least 5 cars for each list.

Your program should be able to reasonably detect error conditions (such as invalid transaction code, and an attempt to return a non-existent car) and print an appropriate error message.