**Programming Principles II**  **Assignment 3**

Write a program to keep records of cars in a car dealer shop. The information of each car contains, ID, make, model, year, color, engine, and price. The program must load/save the data about cars from/to a text file on the hard disk (e.g, "c:\\Users\YourName\\cars.txt". Once started it should load the file into array of structs The program should prompt the user to select the operation from a menu as shown below.

1. Add a car record
2. Delete a car record
3. Update a car record
4. view all car records
5. view cars with price less than value
6. view cars by car make
7. view all cars sorted by price
8. view all cars sorted by year
9. save
10. quit

The easiest way to solve this problem is to divide it into steps.

1) Declare a structure called Car to store the cars records. The structure should contain 7 fields:

<id:char[20]><make:char[20]><model:char[20]><year:int><color:char[20]>

<engine:int><price:int>

struct car { int id; char [20] make; ...

...

};

or use typedef

typedef struct { int id; char [20] make; ...

...

} Car;

1. Write a function menu that displays the menu items described above.

1. Write a function updateRec that takes the array of cars and its size as parameters, prompts the user to enter a car ID to update, then your function should check if the record exists. If so, it should prompt the user to enter a new value for each field. You should save these values in a temporary structure. At the end you should display the new record and ask the user to confirm whether to commit the update or to cancel.

1. Write a function addRecord that takes as parameter array of car records and its size. First the function should make sure that it can add new record by checking that the array size is less than MAXSIZE. Then it should prompt the user for the values of the car record. After reading the data and save it in a temporary structure, you should display and ask the user to confirm the addition. Note since this function alter the size of the array, the size should be passed by reference.

1. Write a function deleteRecord, the function should prompt the user to enter the id of a car record to be deleted. Then it should search for the record and remove it from the array. You can remove the record from the array by copying the last record over the record to be deleted and then and then remove the last record by just decrementing the size of the array by 1. Before you remove the record you should display it and ask the user to confirm the deletion. Note since this function alter the size of the array, the size should be passed by reference.

1. Write a function display that takes the array of cars and its size as parameters and display all records.

1. Write a function displayByPrice that takes the array of cars and its size. The function should prompt the user to inter a price and the function should display all cars that has a price lower than that price.

1. Write a function displayByModel that takes the array of cars and its size. The function should prompt the user to enter a make and then it should display all cars that have that make. For example, if the users enter Honda, your code should display all honda cars.

1. Write a function displayByYear that takes the array of cars and its size. The function should prompt the user for a year and after reading the year it should display all cars manufactured in that year.

10)Write a function displaySortedPrice, that takes the array of cars and its size as parameters and then displays all cars sorted according to their price from lowest to highest price.

11)Write a function save that takes the array of cars and its size and a file name. The function should overwrite the file if it exists and store the array of cars information in the file. The first line of the file should contain the number of car records N to be saved, followed by N line where each line contain a record of car in the following format:

<id ><make><model > <year > <color > <engine ><price>

Note your program once started should tries to open the cars.txt file located in a known location on you hard disk(say C:\\Users\Name\cars.txt) If the file doesn't exist, then you have no cars and create a file if the user added new cars.

Save the following data in the cars.txt file and use it for testing.

5

1. honda civic 2002 red 1400 70000
2. ford torus 2005 white 1800 60000
3. honda accord 2001 black 1600 60000
4. kia forte 2001 blue 1800 45000
5. kia sorato 2001 silver 1600 42000