**INTRODUCTORY CLEARSPEED PROBLEM**

**I. CAR LOT DATABASE PROBLEM**

You will be provided with a headstart file, carlot.tar, which contains the following files:

* asc.cn
* asc.h
* carlot.h
* carlot\_next.cn

The two header files, asc.h and carlot.h, will need to be included in your program. The file asc.cn contains implementations of some important associative functions that you will need for all the problems involving associative programming on ClearSpeed. The fourth file, carlot\_next.cn, provides examples of solutions similar to the functions you will need to implement for this problem. These files are posted on my password protected ClearSpeed site in the table titled “ClearSpeed Associative Functions. An improved Version 2 of the files in this table have been developed by Kevin Shaffer and will be posted in a separate table following the earlier Version 1. Please use Version 1 to get started, but switch to Version 2 as soon as it is posted on my ClearSpeed site.

For further information, refer to the “Introductory Programming Manual”. It is probably the best one for beginning Cn programmers. A more advanced ClearSpeed Cn programming guide is Chapter 11 in the “SDK Reference Manual”, which provides more detailed information. You are free to use any of the ClearSpeed reference manuals or instruction slides.

**II. REQUIREMENTS**

You will write a program my\_carlot.cn that answers the following questions or performs the following tasks. These questions are based on the carlot database, which is generated by a call to the “poly struct Cars carlot(poly struct Cars car)” function defined in carlot.h (see carlot\_next.cn about how to generate the database).

.

1. Print the first car lot list generated by the carlot.h program under the heading of “Original Car Lot”.
2. Starting with the above data base, change the color of the first 1992 car to ‘R’ and the color of the third 1992 car to ‘L’. Afterwards, print out the resulting car lot list under the heading of “Second Car Lot”
3. Starting with the second car lot data base, make the following changes: If any 1992 car’s color is ‘R’, change ALL of the onlot status of 1992 cars to ‘0’. Afterwards, print the resulting car lot list under the heading of “Third Car Lot”
4. Starting with the third car lot, if there is no 1991 car whose color is ‘A’, change all of the onlot status of 1991 cars to ‘0’. Afterwards, print out the resulting list under the heading of “Fourth Car Lot”.
5. Compute and print a message giving the number of cars satisfying the following conditions: latest model Ford cars that are red and on the lot. Use the ***initial car lot***.
6. Modify above program to compute and print a message giving the number of cars satisfying the following conditions: oldest model Toyota cars that are black and on the lot. Use the ***initial car lot***.
7. Locate and count the number of black cars that are on the initial car lot. If there are more than 3 black cars, change the color of the newest two cars to blue. Use *the initial car lot* and print the resulting car lot under the heading of “Fifth Car Lot”