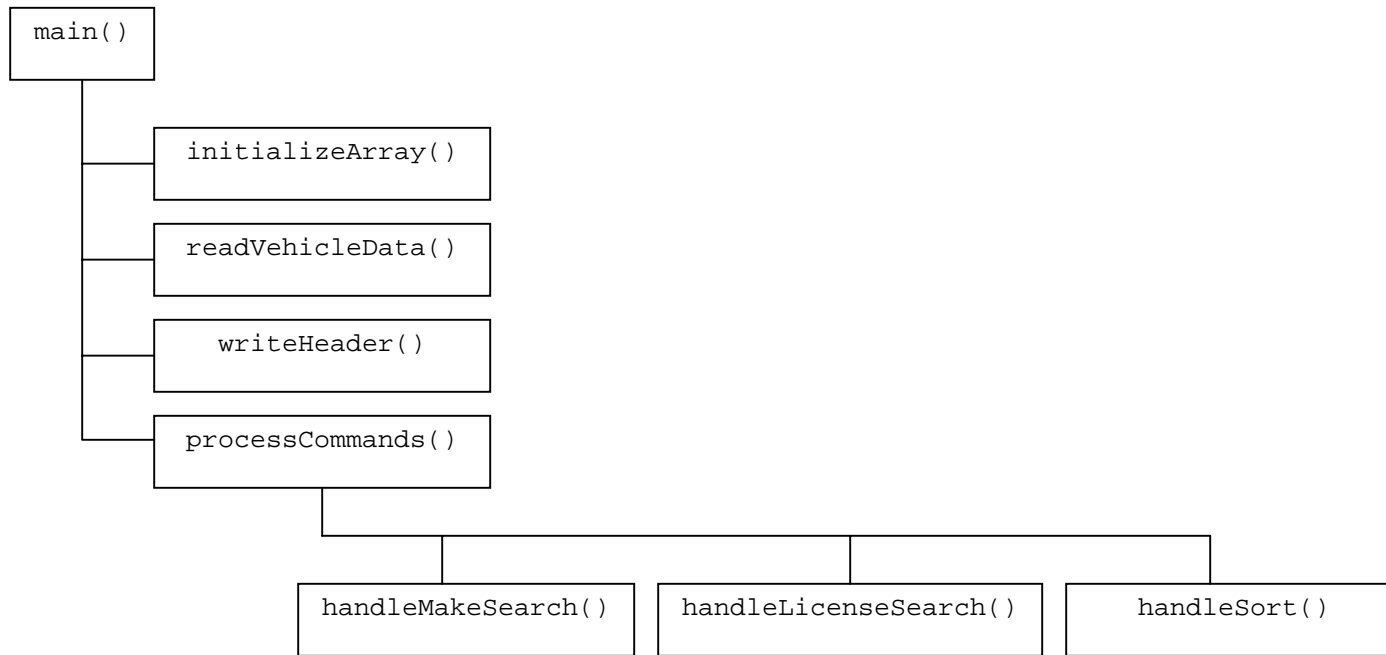


Project 8 Design
Bill McQuain

Structure Chart:



This is a fairly minimal design. There are a number of good candidates for additional functions. For example, it would make sense to have a "helper" function that `handleMakeSearch()` would call to perform the actual search of the array, separating reading the command parameter (the manufacturer name) from the search logic. A similar case can be made for having a helper function to carry out the license search, and even for a separate sort function to manage the actual sorting.

Individual Function Design Outlines:

```
// Function to read and store vehicle data:
I.   Skip everything up to and including a colon character.

II.  Extract the number of vehicles, numVehicles.

III. Set vehicleCount to 0.

IV.  Try to read data for the first vehicle:
     A. Read the license number, discarding the first tab, into a temp variable.
     B. Read the manufacturer name, discarding the second tab, into another temp var.
     C. Read the model name, discarding the third tab, into another temp var.
     D. Extract the year into another temp variable.
     E. Get rid of the newline character.

V.   While vehicleCount < numVehicles and no input failure has occurred:
     A. Store the data just read in the vehicle data array at index vehicleCount:
     B. Increment vehicleCount.
     C. Try to read data for another vehicle:
         i.  Repeat step IV.A
         ii. Repeat step IV.B
         iii. Repeat step IV.C
         iv. Repeat step IV.D
         v.  Repeat step IV.E

VI.  Return the number of vehicles for which data was read.

// The while loop logic used above could be replaced with a for loop without
// the priming read. That would risk an error if the input file was incorrect,
// but would meet spec.

// If a priming read IS used, then the use of temporary variables to store the
// values when they are read is NOT optional. If you read directly into the array
// in that case, you will corrupt the values stored in the first unused cell (when
// the array was initialized earlier).
```

```
// Function to handle a make search command:
I.   Read the manufacturer name, Target.

II.  Get rid of the newline character.

III. Set a Boolean flag Found to false.

IV.  For Index = 0 to numVehicles - 1:
      A. If Target is the same as the make name for the vehicle at Index:
         i.  Write Index.
         ii. Write a colon character.
         iii. Write the license for the vehicle at Index.
         iv. Set Found to true.

V.   If Found is false write that the manufacturer wasn't found.

VI.  Write a newline character.

// The only subtle thing here is that you must have some way of knowing, after the
// search loop in step IV has completed, whether or not any matches were found.
// The simplest way is to use a Boolean flag, but you could also count the number
// of matches, which might be useful.
```