**CSCI 1411 – Lab 08 – Searching and Sorting Arrays**

**Goals:**

* Searching through arrays
* Sorting arrays with bubble and selection sorts

**Development Environment:** (all students must use Visual Studios)

**Skills:** arrays, searching sorting, functions,

**Reading**: Chap 7, Chap 8

**Deliverables:** 1) This lab with two screen shots 2) lastnameFirstLab08.cpp

**Part I – Skills Practice (10 points)**

* Open a new project in Visual Studios. Call the project Lab08. Don’t forget to uncheck the Pre-compiled headers and Security Development Lifecycle checks. Check the box for an empty project

You will be creating 4 files called lab08a.cpp, Car.h and CarFunctions.h, CarFunctions.cpp

In Car.h. If you start with the working files from a working Lab07, you will have a lot less to type.

Create a file called Car.h under header files

**#ifndef CAR\_H**

**#define CAR\_H**

**#include <string>**

**using namespace std;**

**struct Car**

**{**

**string make;**

**string model;**

**int year;**

**};**

**#endif //CAR\_H**

Remember in your Car.h, you should remove the #pragma once, and add the duplicate guard:

* **Now** create a file called **CarFunctions.h** under header files

**#ifndef CARFUNCTIONS\_H**

**#define CARFUNCTIONS\_H**

**#include "Car.h"**

**Car newCar(istream &in);**

**bool addCar(Car CarToAdd, Car carArray[], int &currentSize, const int MAXSIZE);**

**void listCars(ostream &out, Car carArray[], int &currentSize);**

**void searchCarMake(string make, Car carArray[], int currentSize);**

**void sortCarMake(Car carArray[], int currentSize);**

**void swap(int&a, int &b);**

**void swap(string &a, string &b);**

**#endif**

* Now create a file called **CarFunctions.cpp** under source files

**#include <iostream>**

**#include "Car.h"**

**#include "CarFunctions.h"**

**using namespace std;**

**Car newCar(istream &in)**

**{**

**Car car1;**

**cout << "Make: ";**

**in.ignore();**

**getline(in, car1.make);**

**cout << "\nModel: ";**

**getline(in, car1.model);**

**cout << "\nYear: ";**

**in >> car1.year;**

**cout << endl;**

**return car1;**

**}**

**bool addCar(Car carToAdd, Car carArray[], int &currentSize, const int MAXSIZE)**

**{**

**currentSize++;**

**if (currentSize < MAXSIZE)**

**{**

**carArray[currentSize - 1] = carToAdd;**

**return true; //successful add**

**}**

**else**

**return false;**

**}**

**void listCars(ostream &out, Car carArray[], int &currentSize)**

**{**

**for (int i = 0; i < currentSize; i++)**

**{**

**out << "Car " << i + 1 << endl;**

**out << carArray[i].make << endl;**

**out << carArray[i].model << endl;**

**out << carArray[i].year << endl << endl;**

**}**

**}**

**void searchCarMake(string make, Car carArray[], int currentSize)**

**{**

**bool found = false;**

**for (int i = 0; i < currentSize; i++)**

**{**

**if (carArray[i].make == make)**

**{**

**found = true;**

**cout << "Car at position " << i << endl;**

**cout << carArray[i].make << endl;**

**cout << carArray[i].model << endl;**

**cout << carArray[i].year << endl<<endl;**

**}//if**

**}//for**

**if (!found)**

**cout << "Record not found\n";**

**}**

**void sortCarMake(Car carArray[], int currentSize)**

**{**

**//bubble sort**

**int maxElement;**

**int index;**

**for (maxElement = currentSize - 1; maxElement > 0; maxElement--)**

**{**

**for (index = 0; index < maxElement; index++)**

**{**

**if (carArray[index].make > carArray[index + 1].make)**

**{**

**//swap the entire record (make, model and year)**

**swap(carArray[index].make, carArray[index + 1].make);**

**swap(carArray[index].model, carArray[index + 1].model);**

**swap(carArray[index].year, carArray[index + 1].year);**

**}//if**

**}//for**

**}//for**

**}//swap**

**void swap(int&a, int &b)**

**{**

**int temp = a;**

**a = b;**

**b = temp;**

**}**

**void swap(string &a, string &b)**

**{**

**string temp = a;**

**a = b;**

**b = temp;**

**}**

* Finally, create a file called **lab08a.cpp**

**#include <iostream>**

**#include "Car.h"**

**#include "CarFunctions.h"**

**using namespace std;**

**int main()**

**{**

**const int MAXSIZE = 100;**

**int currentSize = 0;**

**Car car1;**

**Car carArray[MAXSIZE];**

**int menu;**

**string make;**

**bool done = false;**

**while (!done)**

**{**

**cout << "1. Add a new car to the array\n";**

**cout << "2. List out cars\n";**

**cout << "3. Search for a car by Make\n";**

**cout << "4. Sort cars by Make\n";**

**cout << "5. exit\n";**

**cin >> menu;**

**switch (menu)**

**{**

**case 1: car1 = newCar(cin);**

**addCar(car1, carArray, currentSize, MAXSIZE);**

**break;**

**case 2: listCars(cout, carArray, currentSize);**

**break;**

**case 3: cout << "What make do you want to search for? ";**

**cin >> make;**

**searchCarMake(make, carArray, currentSize);**

**break;**

**case 4:**

**sortCarMake(carArray, currentSize);**

**break;**

**case 5: exit(0);**

**break;**

**default: cout << "Number between 1 and 5\n";**

**}//switch**

**}//while**

**}//main**

* **Run** it adding two cars. Take a **screen shot** of the successful output and place it below. Replace this output with your output (and 2 different cars) For a Windows 10 screen shot: Alt key + PrtSc key. Then Ctrl + V to paste. For Mac: Shift + Command + 4. You will not credit unless you have a successful screen shot with Your name in the output.

 