Lab 5 - section F

Objectives:
In this lab, you need to modify your functions createArray, getArraySize, and freeArray based on the given struct definition below:

```
typedef struct {
   int empID, age, ssn;
   float salary;
} Record;
```

In createArray function, a Record array needs to be created with its size and the maximum SSN of all employees stored in front as two integer numbers. After creating the array, your array should look like this:

size	Max SSN	Emp[0]	Emp[1]	...	Emp[n-1]

You also need to modify the other two functions accordingly. In this lab, employee records should be read in main program, not createArray function.

Main program steps:
1. Open the given csv file
2. Read an integer giving the number of employee records from the first line of the csv file.
3. Allocate an array of employee records (structs) to hold the records in the file like described before.
4. Read the records into the allocated array. Each of the lines in file contains the salary, the age , the ID, and the SSN for one employee.
5. Print out the size of the array, the maximum SSN among all employees, and the information of the last employee in array.
6. Free the created array using your “freeArray” function.

Example output:
Array size is 10000, and the maximum SSN is 999566237
Last employee has ID 9979, age 60, salary 77930.29, and SSN 625407867

Grading Criteria:
Main program: 10 points

createArray function: 10 points

getArraySize function: 5 points

freeArray function: 5 points

Note:

1. Uncomment the first line in starter code if you are using Visual Studio for this lab.
2. Please place “employee.csv” in the location where you have the source files.
   For example: Place “employee.csv” in the folder where you have the source.c file
3. To compile strictly with C flags, please follow the following instruction when you create your project in VS:

VS -> new project -> empty project -> source files -> Add -> Add new item -> Source.c

Please make sure that your source file is in “.c” extension, NOT “.cpp” extension.

General note:

1. If your code does not compile, you will receive an automatic 0 for this assignment.
2. Changing the given function prototype or struct definition will lead to an automatic zero grade.
3. Using any global variables will lead to an automatic zero grade.
4. The implementation of the function should include comments describing what it is intended to do and how this function should be called. Example can be found in CS 2050 lab policy.
5. If your submission does not include a source file, you will receive an automatic zero grade.