

COMP10120 Practical Set 3: Arrays, Duration and Scope

Please read the questions carefully. Name each program based on your student number, the practical set number and question number. For this set (set3), question 1 should be named 1234567s3q1.c where your student number replaces 1234567. All questions that you are submitting can be zipped into a single file called 1234567s3.zip, where 1234567 is your student number and s3 refers to set 3. This zipped file can be submitted via Moodle for grading.

Part 1

1. Write a C program which is used to keep track of the number of people passing a turn style at the local stadium. The program should have four functions, one which is called when a person enters the stadium and one which is called when a person leaves the stadium. When a person leaves, they cannot re-enter the stadium. It costs €20 for a ticket to the concert so a function should keep a track of total amount of money that has been taken and print that to the screen. The final function should report the current number of people at the concert. Provide test code in the main function to demonstrate the program. **The program must demonstrate the use of a global variable and a local static variable.**

Part 2

2. Write a C program which initially creates 2 two-dimensional integer arrays: *arrayA* and *arrayB*. The size of the arrays should be equal and specified by user input (maximum 20 rows and 20 columns). Each array should be filled with random numbers between 100 and 1500 with **no duplicate numbers** in a single array. The program should then create new arrays for each of the following situations:
 - An array containing the product of corresponding **cells** in *arrayA* and *arrayB*
 - An array containing the sum of corresponding **cells** in *arrayA* and *arrayB*
 - An array containing the sum of corresponding **rows** in *arrayA* and *arrayB*
 - An array containing the sum of corresponding **columns** in *arrayA* and *arrayB*

The program should also

- Print the contents of *arrayA* and *arrayB*.
 - Print the number of total number of numbers that occur in both arrays (if any)
 - Print the word *Wow* if the same number appears in the same cell in the two arrays.
3. Write a C Program which prompts the user to enter the size of 2 two-dimensional arrays (they can be different dimensions). The user should also be requested to fill the arrays. The program should then compute the product of the arrays using matrix multiplication rules and print the resulting array. **Note: If the two matrixes are such that they can't be multiplied then an error message should be displayed on the screen^{1,2}.**

¹<http://www.mathwarehouse.com/algebra/matrix/multiply-matrix.php>

²<http://www.mathwarehouse.com/algebra/matrix/multiply-matrix.php#how-do-we-multiply-two-matrices>

4. You work in an insurance company, you have installed trackers into several of your customers' cars in order to monitor their driving and offer a discount if they are 'good'

drivers. Write a C Program that creates a two-dimensional array to record the average daily acceleration of 5 of your customers over the past 10 trips. The data are shown below. The program should prompt you (the user) for a driver number and then return the data of that driver along with a summary of the data and some advice regarding the discount. If the total average acceleration is more than 5 or more OR any single trip average is more 6 or more, then no discount should be applied to the customer. For example, if the user enters 1 the program responds with:

Driver 1 Summary:

Acceleration Recordings: 3,1,3,4,6,7,3,2,5,3

Max average acceleration for Driver 1 was in trip 7

Average acceleration over all of Driver 1 trips: 3.7

Discount: No Discount to be applied

The average acceleration of all drivers over all trips is: 3.46

Driver data:

Driver 1 – 3, 1, 3, 4, 6, 7, 3, 2, 5, 3

Driver 2 – 2, 2, 2, 3, 4, 4, 3, 3, 3, 2

Driver 3 – 5, 6, 7, 6, 5, 3, 3, 3, 3, 3

Driver 4 – 3, 3, 3, 3, 4, 4, 3, 2, 1, 3

Driver 5 – 2, 3, 4, 6, 6, 2, 5, 1, 3, 3

The units of acceleration in this case are mph per second.