

Computer Programming

Lab10

May 30, 2025

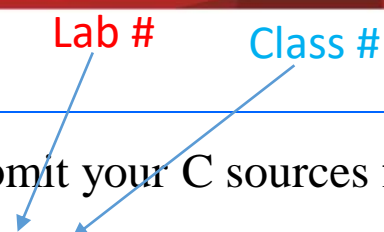


Submission

- **Submit to server**

At the end of the Lab10, submit your C sources file by typing

`~gs1401/bin/submit Lab10_5 ex10_3.c ex10_extra1.c ex10_extra2.c // by Fri. 13:50`



You may check that you have submitted your source code correctly by typing

`~gs1401/bin/submit -check`

Ex3

- Write a program that reads two integers N and M , representing the number of rows and columns of a 2D integer array. Then read $N \times M$ integers to fill the array. Your task is to find the maximum value in the array and print this value followed by all positions (row and column indices) where this maximum value appears.
 - Input:
 - ✓ First line contains two integers N and M ($1 \leq N, M \leq 100$).
 - Output:
 - ✓ Row and column indices start at 0.

- Program output

```
[ohyong@cse ~/cp/Lab10]$ vi ex10_3.c
[ohyong@cse ~/cp/Lab10]$ gcc ex10_3.c -o ex10_3
[ohyong@cse ~/cp/Lab10]$ ./ex10_3
Enter dimensions (N M): 2 3
Enter values:
4 9 7
9 2 5
Max value: 9
Positions: (0, 1) (1, 0)
```

```
[ohyong@cse ~/cp/Lab10]$ ./ex10_3
Enter dimensions (N M): 3 4
Enter values:
1 5 9 3
9 7 9 2
4 6 1 9
Max value: 9
Positions: (0, 2) (1, 0) (1, 2)
(2, 3)
```

- Write the following program.
 - ✓ Declare an int array of size 10 and input a value.
 - ✓ **Implement** the max function, pass the array, return the maximum value, and print that value in main().
 - `int max(int arr[], int size);`
 - ✓ **Implement** the avg function, pass the array, return the average value, and print that value in main().
 - `double avg(int arr[], int size);`

- **Program output**

```
[ohyong@cse ~/cp/Lab10]$ vi ex10_extra1.c
[ohyong@cse ~/cp/Lab10]$ gcc ex10_extra1.c -o ex10_extra1
[ohyong@cse ~/cp/Lab10]$ ./ex10_extra1
Enter 10 integers:
Number 1: 34
Number 2: 23
Number 3: 45
Number 4: 67
Number 5: 12
Number 6: 78
Number 7: 89
Number 8: 90
Number 9: 21
Number 10: 55
Maximum value in array: 90
Average value in array: 51.40
```

Extra2

- Write a program that reads a 3×3 matrix of integers. Determine if this matrix is a magic square: all rows, columns, and both main diagonals must have the same sum.

- **Program output**

```
[ohyong@cse ~/cp/Lab10]$ vi ex10_extra2.c
[ohyong@cse ~/cp/Lab10]$ gcc ex10_extra2.c -o ex10_extra2
[ohyong@cse ~/cp/Lab10]$ ./ex10_extra2
Enter 9 values for 3x3 matrix:
8 1 6
3 5 7
4 9 2
Magic Square

[ohyong@cse ~/cp/Lab10]$ ./ex10_extra2
Enter 9 values for 3x3 matrix:
1 2 3
4 5 6
7 8 9
Not a Magic Square
```