

CSE108 – Computer Programming Laboratory

(Spring 2021) Lab #5

Hand-in Policy: A file named *StudentNo.c* should be submitted via Teams.

Collaboration Policy: No collaboration is permitted.

Grading: This lab will be graded on the scale of 100 [*Compilation & Running (30pts), Question 1 (40pts), Question 2 (10pts), Style and Comments(20pts)*]

1. The program should calculate the Sigma Notation for three expressions below:

$$\sum_{x=0}^n x^2 + 5$$

(Expression 0)

$$\sum_{x=0}^n 2x + 1$$

(Expression 1)

$$\sum_{x=0}^n x^2$$

(Expression 2)

The number “n” should be defined by user. After that the program calculates the summation of each expression (“n” is same for all of them) and print them. The program should hold the following rules:

- The number “n” should be defined by user (it is same for each expression).
- Make sure the program includes “n” to the calculations. ($0 < x \leq n$)
- The program should calculate the result of each expression and keep them in a 1d array.
- The program should pass the array to a function which writes the results on it.
- There should be 3 functions for the calculation of each expression (these three should be called by using function as a function parameter, not manually) and also a function for addition.
- Every function should be “void” (you should use pointers).

Expected output:

```
n: 3
Sum of expression0: 34
Sum of expression1: 16
Sum of expression2: 14
```

Use the following function prototypes:

```
void expression0 (int x, int *result){
    /* calculation of the expression0 for the value of x*/ }
void expression1 (int x, int *result){
    /* calculation of the expression1 for the value of x*/ }
void expression2 (int x, int *result){
    /* calculation of the expression2 for the value of x*/ }
void addition (void (*func)(int, int *), int *sum, int n){
    /* obtaining the calculations for a single expression and return the result */ }
void calculation (int *sumArray, int n){
    /* Obtaining the summations and assinging them to sumArray respectively*/ }
int main (){
    /* Declaring array, reading “n”, obtaining the results by using functions, and printing them.
    Nothing else.*/ }
```

2. The program creates a 2 dimensional array with 10 rows and 10 columns. Each element of the this 2d array (aka matrix) should be assigned with a 2-digit integer randomly. To obtain a random 2-digit integer you can use `rand()` function as follows (do not forget to include `stdlib.h` to use it):

```
int randomNumber = rand () %90 + 10;
```

The creation of array should be handled in a function with the following prototype:

```
void createArray(int (*array)[10])
```

Once the matrix is created, it should be printed to the screen. After that the program should reach the spesific elements of the matrix which selected by user and print them (program should do this again and again until an invalid input). User should enter 2 integers to point the element. If one of the inputs are invalid, the program should raise an error and terminate.

Expected output:

```
Matrix:

83 26 37 35 33 35 56 22 79 11
42 77 60 89 33 86 70 46 62 56
81 18 97 79 72 90 72 23 47 65
69 82 82 58 69 67 83 26 41 62
79 73 91 39 24 77 78 84 75 40
93 56 11 90 36 73 42 60 86 41
25 55 24 97 66 45 26 49 23 57
64 55 82 55 84 97 84 24 43 60
17 88 16 18 78 94 43 21 54 29
52 70 36 28 29 92 26 46 94 39

Which element of the matrix do you want to reach?
i: 5
j: 6
5. row 6. column of the matrix is 42

Which element of the matrix do you want to reach?
i: 0
j: 0
0. row 0. column of the matrix is 83

Which element of the matrix do you want to reach?
i: 10
j: -1
Invalid input. Terminating...
```

P.S. Perform these tasks respectively in a single program.