CL-1004
Object Oriented Programming Lab
Lab # 02

Objectives:
- Pointers
- Pointer Variable Declarations and Initialization
- Referencing/Dereferencing, Pointer Arithmetic Pointers & Functions

1. There must be a block of comments at start of every question's code by students; the block should contain brief description about functionality of code.
2. Comment on every function about its functionality.
3. Use understandable name of variables.
4. Proper indentation of code is essential
5. Write a C++ statement(s) for each of the following task one after the other, in the same order.
6. Make a Microsoft Word file and paste all of your C++ code with all possible screenshots of every **task output in MS word and submit .cpp file with word file**.
7. Make separate .cpp files for all tasks and use this format **22F-1234_Task1.cpp**.
8. First think about statement problems and then write/draw your logic on copy.
9. After copy pencil work, code the problem statement on MS Studio C++ compiler.
10. At the end when you done your tasks, attached C++ created files in MS word file and make your submission on Google classroom. (Make sure your submission is completed).
11. Please submit your file in this format **22F-1234_L1**.
12. Do not submit your assignment **after deadline**.
13. **Do not copy code from any source otherwise you will be penalized with negative marks.**

Problem 1: | Pointer Arithmetic

Write a program which declares an array of 5 numbers. Initialize the array with 3, 9, 1, 15, 3. Now use the

- **Increment operator**
- **Decrement operator**

in pointers to print the address and values of each index. Also compare all values of pointers

and display if some values are equal.

Problem 2: | Pointer with function

Write a program for passing pointer to a function primaryCheck() while returns true if number is prime otherwise return false.

Problem 3: | Double pointers

Write a C++ program where you have two integer variables

```
int firstvalue = 15, secondvalue = 25;
```

and four pointers

```
int * p1, * p2, **p3, **p4;
```
Use p3 and p4 to store the address of p1 and p2

You have to perform the following steps

- `// p1 = address of firstvalue`
- `// p2 = address of secondvalue`
- `// p3 = address of firstpointrer`
- `// p4 = address of secondpointer`
- `// value pointed by p1 = 100`
- `// value pointed by p2 = value pointed by p1`
- `// p1 = p2 (address of pointer is copied or not)`
- `// p3 =p4  (check address of pointer is copied or not)`
- `// value pointed by p1 = 200`
- `// print firstvalue, secondvalue`

And comment like above after each step.

Problem 4: | Pointers

For each of the following, write C++ statements that perform the specified task. Assume that double-precision, floating-point numbers are stored in eight bytes and that the starting address of the array is at location 1002500 in memory. Each part of the exercise should use the results of previous parts where appropriate.

(a) Declare an array of type double called numbers with 10 elements, and initialize the elements to the values 0.0, 1.1, 2.2, . . . , 9.9. Assume that the symbolic constant SIZE has been defined as 10.

(b) Declare a pointer nPtr that points to a variable of type double.

(c) Use a for statement to print the elements of array numbers using array subscript notation. Print each number with one position of precision to the right of the decimal point.

(d) Write two separate statements that each assign the starting address of array numbers to the pointer variable nPtr.

(e) Use a for statement to print the elements of array numbers using pointer/offset notation with pointer nPtr.

(f) Use a for statement to print the elements of array numbers using pointer/offset notation with the array name as the pointer.

(g) Use a for statement to print the elements of array numbers using pointer/subscript notation with pointer nPtr.

(h) Refer to the fourth element of array numbers using array subscript notation, pointer/offset notation with the array name as the pointer, pointer subscript notation with nPtr and pointer/offset notation with nPtr.

(i) Assuming that nPtr points to the beginning of array numbers, what address is referenced by nPtr + 8? What value is stored at that location?

(j) Assuming that nPtr points to numbers[5], what address is referenced by nPtr after nPtr-= 4 is executed? What's the value stored at that location?

**Problem 6:	Pointers**

Define an **int*** pointer variable **a**. Then:

  (1) Use **new** to make **a** point to a dynamic array of **5** cells of type **int**.
  (2) Write a loop to fill **a** with values 3, 7, 11, 15, 19.
  (3) Using **Print function** to print values stored in **a**.

**Problem 5:	Double pointers with function**

Write a C++ program to display a matrix of size 3x3.
     Passing double pointer to a function print() and display matrix.

Proper code indentation will hold extra marks!

Best of luck 😎

You are done with your exercise, submit to Classroom at given time.