

# Assignment # 1

**Course Title:** Object Oriented Programming

**Due Date:** 13/03/2022

---

## Instructions:

1. Assignments are to be done individually. You must complete this assignment by yourself. You cannot work with anyone else in the class or with someone outside of the class. The code you write must be your own and you must understand each part of coding.
2. The AIM of this assignment is to practice with Pointers, Dynamic Memory Allocation (DMA) in C++ and recursion.
3. Plagiarism of any kind (copying from others and copying from internet, etc.,) is not allowed and will be severely penalised.
4. Submission: Submit the code for each question as a separate .cpp file. Put all the .cpp files in a single folder and zip it prior to submission. Name your folder as Assignment1\_RollNumber.zip. There should be no subfolders within the main folder. Not following submission instructions will result in deduction of marks.

## Question 1.

Write a function to swap the values of two integer variables by using pointers (instead of reference variables) and return its swapped values to the main function. (3 points)

## Question 2.

Write a piece of code that declares an integer array of size 10. Take 10 different integers from user and print the difference of maximum and minimum array elements. You must use pointer to access/manipulate array elements.

## Question 3.

Write a function whose signature looks like (char\*, char). This should search in the first argument for the character passed as second argument. If it finds the character it should return 1 else 0.

## Question 4.

Consider two arrays arr1 and arr2 as shown above. You need to write a program to combine these two arrays into arr3. You need to use pointers to access/manipulate the array elements. You need to keep following conditions in mind:

Compare the elements in first location, since 1 is less than 3 so you will write 1 first to the arr3 and increment pointer for arr2. Next you will compare 3 with 5, since 3 is less so you will copy 3 into arr3's 2nd location and increment pointer for arr1, so and so on. Since arr2 is larger than arr1, the last remaining elements will be copied as it is to arr3.

arr1

3	2	3	6	8
---	---	---	---	---

arr2

1	5	0	4	7	9	10	11
---	---	---	---	---	---	----	----

arr3

1	3	2	3	5	0	4	6	7	8	9	10	11
---	---	---	---	---	---	---	---	---	---	---	----	----

### Question 5.

Write a C++ program that outputs a histogram of student Marks for a Mid-term-1 Examination. The program should input each student's Marks as an integer and store the Marks in a dynamic array. Marks should be entered until the user enters -1 for marks. The program should then scan through the Dynamic array and compute the histogram. In computing the histogram, the minimum value of a marks is 0 but your program should determine the maximum value entered by the user. Then use a dynamic array to store and output the histogram. (20 points)

For example, if the input is:

80  
60  
80  
70  
60  
50  
50  
50  
-1

Then the output should be:

The frequency of 80's: 2  
The frequency of 70's: 1  
The frequency of 60's: 2  
The frequency of 50's: 3

### Question 6.

Take name as input from user. Write a recursive function to print the name in reverse order.

### Question 7.

Write a program that takes as input the length of the top line and prints a Z pattern with asteriks using recursion. For example, if input = 5, it should print:

```
* * * * *
      *
      *
      *
    *
  * * * * *
```

**Question 8.**

Write a recursive program to merge two sorted, equal length arrays such that the resulting array is also sorted. For example if the following arrays are given:

Array 1: 2,4,6,7,9,10

Array 2: 3,5,8,11,20

The output should be:

2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 20

**Question 9.**

Write a matrix multiplication program that first get the dimensions of two matrices as input from the user, then gets their elements from the user, and then uses a recursive function to multiply the matrices. Global and static variables are not allowed. The program must check if the matrix dimensions are valid for multiplication; if not, it should continue to ask the user to input new matrices of the correct dimensions.

**Question 10.**

Write a recursive program that takes an input N and outputs all possible positive integer combinations that add up to N, in decreasing order of length. For example, if  $N = 3$ , the output will be

1 1 1

1 2

3

Your program should work for any positive integer value of N.