Sr.	Program
1	Given **two sorted arrays** of sizes **m** and **n** respectively, the task is to find the element that would be at the **k-th position** in the final sorted array formed by merging these two arrays. **Input**: a[] = [2, 3, 6, 7, 9], b[] = [1, 4, 8, 10], k = 5 **Output**: 6 **Explanation**: The final sorted array is [1, 2, 3, 4, 6, 7, 8, 9, 10] The 5th element is 6. **Input**: a[] = [100, 112, 256, 349, 770], b[] = [72, 86, 113, 119, 265, 445, 892], k = 7 **Output**: 256 **Explanation**: The final sorted array is [72, 86, 100, 112, 113, 119, 256, 265, 349, 445, 770, 892]. The 7th element is 256.
2	Given two strings **s1** and **s2** consisting of **lowercase** characters, the task is to check whether the two given strings are **anagrams** of each other or not. An anagram of a string is another string that contains the same characters, only the order of characters can be different. **Input**: s1 = “geeks” s2 = “kseeg” **Output**: true **Explanation**: Both the strings have the same characters with same frequency. So, they are anagrams. **Input**: s1 = “allergy” s2 = “allergic” **Output**: false **Explanation**: Characters in both the strings are not the same. s1 has extra characters ‘y’ and s2 has extra characters ‘i’ and ‘c’, so they are not anagrams.

3	Given the number of rows and columns, print the corresponding swastika pattern using loops.
	
	Note: The number of rows and columns should be the same and an odd number. This will generate a perfect swastika pattern.
	
	`Input : row = 7, column = 7`
	`Output:`
	`*  * * * *`
	`*  *`
	`*  *`
	`* * * * * * *`
	`  *  *`
	`  *  *`
	`* * * *  *`

4	Given an **integer n**, the task is to find the **first n rows** of Pascal's triangle. **Input** : N = 5 **Output**: 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1
5	Find the difference between the second largest element and the second smallest element of an array. **Input** : Enter the size of array: 7 Enter 7 elements: 5 1 9 7 1 5 3 **Output**: Difference: 4

6	Given the principal amount (P), rate of interest per annum (R), and loan tenure in months (N), **write a program to calculate the EMI (Equated Monthly Installment)** for a loan. **Input :** Enter principal amount (P): 500000 Enter annual interest rate (R): 7.5 Enter loan tenure in months (N): 60 **Output:** The EMI per month is: **10013.05**
7	Given a number n then print n terms of fibonacci series in reverse order. **Input** : n = 5 **Output** : 3 2 1 1 0 **Input** : n = 8 **Output** : 13 8 5 3 2 1 1 0
8	Given an array **arr[]**, the task is to **reverse** the array. Reversing an array means **rearranging** the elements such that the **first** element becomes the **last**, the **second** element becomes **second last** and so on. **Input**: arr[] = {1, 4, 3, 2, 6, 5} **Output**: {5, 6, 2, 3, 4, 1} **Input**: arr[] = {4, 5, 1, 2} **Output**: {2, 1, 5, 4}
9	Write a program that prints the Fibonacci series up to a given number of terms using recursion. **Input**: n = 5 **Output**: 0 1 1 2 3 **Input**: N = 7 **Output**: 0 1 1 2 3 5 8

10	Given a string **S** and a character **'c'**, the task is to count the occurrence of the given character in the string. **Input**: S = "geeksforgeeks" and c = 'e' **Output**: 4 **Explanation:** 'e' appears four times in str. **Input**: S = "abccdefgaa" and c = 'a' **Output**: 3 **Explanation**: 'a' appears three times in str.
11	Find if a given string can be represented from a substring by iterating the substring "n" times. **Input**: str = "abcabcabc" **Output:** true **Input:** str = "aabaabaabaab" **Output:** true **Input:** str = "abcdabc" **Output:** false
12	Given a string **s** representing an expression containing various types of brackets: {}, (), and [], the task is to determine whether the brackets in the expression are balanced or not. A balanced expression is one where every opening bracket has a corresponding closing bracket in the correct order. **Input**: s = "[{()}]" **Output**: true **Explanation**: All the brackets are well-formed. **Input**: s = "([]" **Output**: false **Explanation**: The expression is not balanced as there is a missing ')' at the end.

13	Given two sorted arrays, the task is to merge them in a sorted manner. **Input**: arr1[] = { 1, 3, 4, 5}, arr2[] = {2, 4, 6, 8} **Output:** arr3[] = {1, 2, 3, 4, 4, 5, 6, 8} **Input:** arr1[] = { 5, 8, 9}, arr2[] = {4, 7, 8} **Output:** arr3[] = {4, 5, 7, 8, 8, 9}
14	Given a string that contains a special character together with alphabets (‘a’ to ‘z’ and ‘A’ to ‘Z’), reverse the string in a way that special characters are not affected. **Input**: a!!!b.c.d,e'f,ghi **Output**: i!!!h.g.f,e'd,cba **Input**: str = “Ab,c,de!$” **Output**: str = “ed,c,bA!$”
15	Find the length of the maximum number of consecutive numbers jumbled up in an array. **Input**: arr[] = {1, 94, 93, 1000, 5, 92, 78}; **Output** : 3 **Explanation:** The largest set of consecutive elements is 92, 93, 94. **Input**: arr[] = {1, 5, 92, 4, 78, 6, 7}; **Output**: 4 **Explanation:** The largest set of consecutive elements is 4, 5, 6, 7.
16	Given an array **arr[]** of n integers and a **target** value, the task is to find whether there is a pair of elements in the array whose sum is equal to **target.** **Input:** arr[] = [0, -1, 2, -3, 1], target = -2 **Output:** true **Explanation:** There is a pair (1, -3) with the sum equal to the given target, 1 + (-3) = -2. **Input:** arr[] = [1, -2, 1, 0, 5], target = 0 **Output:** false **Explanation:** There is no pair with sum equals to given target.

17	Given an array **arr[]**, the task is to find the subarray that has the maximum sum and return its sum. **Input**: arr[] = {2, 3, -8, 7, -1, 2, 3} **Output:** 11 **Explanation:** The subarray {7, -1, 2, 3} has the largest sum 11. **Input**: arr[] = {5, 4, 1, 7, 8} **Output**: 25 **Explanation**: The subarray {5, 4, 1, 7, 8} has the largest sum 25.
18	Write a program to merge two arrays into one, but if any element is repeated in the arrays, it should appear only once, and the duplicates should be replaced with zeros. **Input:** Enter the size of first array: 4 Enter the elements of first array: 7 2 5 9 Enter the size of second array: 5 Enter the elements of second array: 5 9 3 7 8 **Output**: Merged array: 7 2 5 9 0 3 8 0 0
19	Given an array of **coins[]** of size **n** and a target value **sum**, where **coins[i]** represent the coins of different denominations. You have an **infinite supply** of each of the coins. The task is to find the **minimum** number of coins required to make the given value **sum**. If it is not possible to form the sum using the given coins, return **-1**. **Input**: coins[] = [25, 10, 5], sum = 30 **Output**: 2 **Explanation** : Minimum 2 coins needed, 25 and 5 **Input**: coins[] = [9, 6, 5, 1], sum = 19 **Output**: 3 **Explanation**: 19 = 9 + 9 + 1 **Input**: coins[] = [4, 6, 2], sum = 5 **Output**: -1 **Explanation**: Not possible to make the given sum.

20	Given a 2D square matrix, find the sum of elements in Principal and Secondary diagonals. For example, consider the following 4 X 4 input matrix. **Input**: 4 1 2 3 4 4 3 2 1 7 8 9 6 6 5 4 3 **Output:** Principal Diagonal: 16 Secondary Diagonal: 20
21	Given a string and write a C program to count the number of vowels and consonants in this string. **Input:** str = "geeks for geeks" **Output:** Vowels: 5 Consonants: 8 **Input**: str = "abcdefghijklmnopqrstuvwxyz" **Output**: Vowels: 5 Consonants: 21
22	Given a number N, the task is to print the prime numbers from 1 to N. **Input:** N = 10 **Output:** 2, 3, 5, 7 **Input**: N = 5 **Output**: 2, 3, 5
23	Given an integer **n**, find whether the number is Palindrome or not. A number is a Palindrome if it remains the same when its digits are reversed. **Input**: n = 12321 **Output**: Yes

	Input: n = 1234 **Output**: No
24	Write a Program to Check Whether a String is a Palindrome or Not. **Input**: s = “level” **Output**: True **Input**: s = “Geeks” **Output**: False
25	Given 2 sorted arrays **a[]** and **b[]**, each of size **n**, the task is to find the median of the array obtained after merging a[] and b[]. **Input**: a[] = [1, 12, 15, 26, 38], b[] = [2, 13, 17, 30, 45] **Output**: 16 **Explanation**: The middle two elements are 15 and 17, so median = (15 + 17)/2 = 16
26	Given a string that contains both upper and lower case characters in it. The task is to count a number of upper and lower case characters in it. **Input**: Introduction to Python **Output**: Lower Case characters : 18 Upper case characters : 2 **Input**: Welcome to GeeksforGeeks **Output**: Lower Case characters : 19 Upper case characters: 3
27	Write a program to sort the numbers in a string expression where numbers are separated by the '+' sign. The program should rearrange the numbers in non-decreasing order while keeping them separated by '+'. **Input**: 3+2+1 **Output**: 1+2+3 **Input**: 1+1+3+1+3 **Output**: 1+1+1+3+3

28	Write a program to print a **center-aligned pyramid pattern** where each row contains numbers increasing from 1 up to the row number and then decreasing back to 1. **Input**: 3 **Output**: 1 121 12321
29	Write a program to find and print the elements that are **non-duplicate** (i.e., unique) between two arrays. An element is considered non-duplicate if it appears in only one of the two arrays, but not in both. **Input**: arr1: 1 2 3 4 arr2: 3 4 5 6 **Output**: 1 2 5 6
30	Given an array of **positive** integers **arr[]** of size **n**, the task is to find the second **largest distinct element** in the array. **Note:** If the second largest element does not exist, return **-1**. **Input:** arr[] = [12, 35, 1, 10, 34, 1] **Output:** 34 **Explanation:** The largest element of the array is 35 and the second largest element is 34. **Input:** arr[] = [10, 10, 10] **Output:** -1 **Explanation:** The largest element of the array is 10 there is no second largest element.
31	Write a program to generate very big (at least 11 digits) and calculate the total of all those numbers. In the program, User will input the first digit i.e. "D", input count i.e. "N" which needs to generate automatically sequential bases from 1 to N, other remaining digits should be 0

	Input Validation $0 \leq D \leq 9$ $1 < N \leq 1000$ Explanation Let's assume the inputs from the user for D is 5 and N is 50 ● So the generated numbers will be 50000000001, 50000000002, 50000000003, 50000000004.... 50000000050. ● The calculated sum of above generated numbers is 2500000001275. ● So output of this program should be 2500000001275. Test Case-1 **Input** D: 5, N: 3 **Output**: 150000000006 Test Case - 2 **Input:** D: 3 **Output** N: 10 Output: 300000000055 Test Case - 3 **Input:** D: 9 N: 100 **Output**: 9000000005050 Note: All data types are allowed except float, double and long in this program.
32	Given an integer array **arr[]** containing digits from **[0, 9]**, the task is to **print all possible letter combinations** that the numbers could represent. A **mapping** of digits to letters (just like on the telephone buttons) is being followed. **Note** that **0** and **1** do not map to any letters. All the mapping are shown in the image below: **Input**: arr[] = [2, 3] **Output**: ad ae af bd be bf cd ce cf **Explanation**: When we press 2,3 then ad, ae, af, bd, … cf are the list of possible words. **Input**: arr[] = [5] **Output**: j k l

	Explanation: When we press 5 then j, k, l are the list of possible words.
33	Given an array of integers **arr[]** of size **n**, the task is to **rotate** the array elements to the **left** by **d** positions. **Input**: arr[] = {1, 2, 3, 4, 5, 6}, d = 2 **Output**: {3, 4, 5, 6, 1, 2} **Explanation**: After first left rotation, arr[] becomes {2, 3, 4, 5, 6, 1}, After the second rotation, arr[] becomes {3, 4, 5, 6, 1, 2} **Input**: arr[] = {1, 2, 3}, d = 4 **Output**: {2, 3, 1} **Explanation**: The array is rotated as follows: After first left rotation, arr[] = {2, 3, 1} After second left rotation, arr[] = {3, 1, 2} After third left rotation, arr[] = {1, 2, 3} After fourth left rotation, arr[] = {2, 3, 1}
34	Given a time in the format of hh:mm (12-hour format) 0 < hh < 12, 0 <= mm < 60. The task is to convert it into words as shown: **Input**: h = 5, m = 0 **Output**: five o' clock **Input**: h = 6, m = 24 **Output**: twenty four minutes past six 5:00 → five o’ clock 5:01 → one minute past five 5:10 → ten minutes past five 5:15 → quarter past five 5:30 → half past five 5:40 → twenty minutes to six 5:45 → quarter to six 5:47 → thirteen minutes to six 5:28 → twenty eight minutes past five

35	Write a program to reverse the elements of an array without using a temporary array. **Input:** Enter the number of elements in the array: 5 1 2 3 4 5 **Output:** Reversed array: 5 4 3 2 1
36	Given two arrays **a[]** and **b[]**, the task is to find the intersection of the two arrays. **Intersection** of two arrays is said to be elements that are **common** in both arrays. The intersection should **not count duplicate elements** and the result should contain items in **any order**. **Input:** a[] = {1, 2, 1, 3, 1}, b[] = {3, 1, 3, 4, 1} **Output:** {1, 3} **Explanation:** 1 and 3 are the only common elements and we need to print only one occurrence of common elements. **Input:** a[] = {1, 1, 1}, b[] = {1, 1, 1, 1, 1} **Output:** {1} **Explanation:** 1 is the only common element present in both the arrays. **Input:** a[] = {1, 2, 3}, b[] = {4, 5, 6} **Output:** {} **Explanation:** No common element in both the arrays.
37	Write a program to print a below pattern. **Input**: 5 **Output**: 1 123 12345 1234567 123456789

38	Write a program to print a below pattern. **Input**: 5 **Output**: 1 121 12321 1234321 123454321
39	Write a program to print a below pattern. **Input**: 5 **Output**: 1 121 1*3*1 1**4**1 1***5***1 **Input**: 7 **Output**: 1 121 1*3*1 1**4**1 1***5***1 1****6****1 1*****7*****1