ARRAY MOCK

- Write a C program to find numbers that occur in exact two times in an array.

- Write a C program to sort an array and print the count of unique elements in the array.

- Write a C program to find smallest number in an unsorted array.

- Write a C program to find the prime numbers in an array and replace it with 0.

- Write a C program to get an array from users and remove odd numbers from the array and find the sum of the existing values in the array.

- Write a C program to skip 2 elements after the occurrence of a non-prime number in an array.

- Write a C program to find the average of all unique elements.

- Write a C program to replace all prime numbers by 0, and even numbers by 1 in an array.

- Write a C program to reverse the array and remove all the multiples of 3 from it.

- Write a C program to delete duplicate elements from an array.

- Write a C program to remove all the prime numbers in an array.

- Write a C program to replace odd numbers with '*' in an array and find the sum of even numbers.

- Write a C program to count even numbers and count prime numbers.

- Write a C program to print all repeating elements in an array.

- Write a C program to put even and odd elements of an array in two separate arrays.

- Write a C program to repeating elements in an array.

- Write a C program to find sum, sum is greater than 100 delete even, sum is less than 100 delete odd numbers.

- Write a C program to read n number of values in an array and display the array after skipping two values next to the multiple of 5.

- Write a C program to reverse the array and delete even numbers in the array.

- Write a C program to find the smallest element in the array without sorting.

- Write a C program to find the frequencies of array elements.

- Write a C program to print all prime numbers in an array.

- Write a C program to print all unique elements in an array.

- Write a C program to remove all the prime numbers in an array.

- Write a C program to find the majority elements in the array (the majority element is the element that appears more than [n/2] times).

- Write a C program to get an array from the users, remove even numbers from the array and   find the average of existing values in the array.

- Write a C program to insert new value in the array.

- Write a C program to read n number of values in an array and display the array after deleting two values next to an odd number

- Write a C program to merge two arrays of same size sorted in descending order.

- Write a C program to delete unique elements in an array.

- In an array replace all the multiple of 3 by '0' and multiples of 5 by'1'.

- Write a C program to find numbers that occur in exact two time in an array.

- Write a C program to find largest number in an unsorted array .

- Write a C program to replace all the multiples of 5 by'10'.

- Write a C program to sort an array and print the count of prime number in the array.

- Write a C program to find the smallest number in an unsorted array.

- Write a C program to sort an array and print the count of unique element in an array.

- Write a C program to find the largest number in an unsorted array.

- Write a C program to count prime number.

- Write a C program to count the frequency of each element of an array.

- Write a C program to an array from the user and remove odd numbers from the array and find the sum of the existing values in the array.

- Write a C program to sort the array and replace element in an even position with '0'.

- Write a C program to delete element from a specified position.

- Write a C program to find the factorial of a largest number in an array.

- Write a C program to delete multiples of 5

- Write a C program to delete lowest element in an array.

- Write a C program to insert an element in a specified position.

- Write a C program to find the largest element in an array without sorting.

- Write a C program to count the total number of unique elements in an array.

- Write a C program to read n number of values in an array, and display It in reverse order.