***CS112\_L\_#03***

**PC LAB**

**CS112\_L\_#03**

HAMZA AHMAD ::2023217

Task#01

HAMZA AHMAD ::2023217#include <iostream>

using namespace std;

void duplicate(int &size, int arry[]) {

    for (int i = 0; i < size; i++) {

        for (int j = i + 1; j < size; j++) { // Corrected the loop control variable to 'j'

            if (arry[i] == arry[j]) {

                // Shift elements to the left to overwrite the duplicate

                for (int k = j; k < size - 1; k++) {

                    arry[k] = arry[k + 1];

                }

                size--; // Reduce size since a duplicate is removed

                j--;    // Adjust the loop counter to recheck the current position

            }

        }

    }

    cout << "Array elements after deleting duplicates: " << endl;

    for (int i = 0; i < size; i++) {

        cout << arry[i] << endl;

    }

}

int main() {

    int size;

    cout << "Enter the size of an array: ";

    cin >> size;

    int \*array = new int[size];

    cout << "Enter the values:" << endl;

    for (int i = 0; i < size; i++) {

        cin >> array[i];

    }

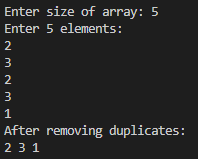
    duplicate(size, array);

    delete[] array; // Free dynamically allocated memory

    return 0;

}

***out put***



Task#02

#include<iostream>

using namespace std;

// Function to create an array of integers and remove duplicates

int\* create\_array(int n) {

    // Dynamically allocate memory for the array

    int\* arr = new int[n];

    // Input elements into the array

    cout << "Enter " << n << " elements:" << endl;

    for (int i = 0; i < n; i++) {

        cin >> arr[i];

    }

    // Removing duplicates

    int newSize = 0; // Variable to track the size of the array without duplicates

    for (int i = 0; i < n; i++) {

        bool duplicate = false; // Flag to check if the current element is a duplicate

        // Check if the current element is a duplicate of any previous element

        for (int j = 0; j < newSize; j++) {

            if (arr[i] == arr[j]) {

                duplicate = true; // Mark as duplicate

                break; // No need to continue checking further

            }

        }

        // If the element is not a duplicate, add it to the modified array

        if (!duplicate) {

            arr[newSize] = arr[i];

            newSize++; // Increment the size of the modified array

        }

    }

    // Print the array after removing duplicates

    cout << "After removing duplicates:" << endl;

    for (int i = 0; i < newSize; i++) {

        cout << arr[i] << " ";

    }

    cout << endl;

    // Return the modified array

    return arr;

}

int main() {

    int size;

    cout << "Enter size of array: ";

    cin >> size;

    // Call the function to create the array and remove duplicates

    int\* arr = create\_array(size);

    // Deallocate the dynamically allocated memory

    delete[] arr;

    return 0;

}

OUT PUT

