Array Operations

1. Insert at Any Index

To insert an element at a specific index, shift all elements from that index to the right and place the new element.

```cpp
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin >> n;
    int a[n + 1];
    for (int i = 0; i < n; i++)
        cin >> a[i];
    int x, val;
    cin >> x >> val;
    for (int i = n - 1; i >= x; i--)
        a[i + 1] = a[i];
    a[x] = val;
    for (int i = 0; i <= n; i++)
        cout << a[i] << " ";
    return 0;
}
```

---

2. Delete from Any Index

To delete an element from a specific index, shift all elements to the left from that position.

```cpp
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin >> n;
    int a[n];
    for (int i = 0; i < n; i++)
```

```
        cin >> a[i];
    int x;
    cin >> x;
    for (int i = x; i < n - 1; i++)
        a[i] = a[i + 1];
    for (int i = 0; i < n - 1; i++)
        cout << a[i] << " ";
    return 0;
}
```

---

3. Reverse an Array

Swap elements from both ends moving towards the center.

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin >> n;
    int a[n];
    for (int i = 0; i < n; i++)
        cin >> a[i];
    for(int i=0;i<n/2;i++)
    {
        swap(a[i], a[n-i-1]);
    }
    for (int i = 0; i < n; i++)
        cout << a[i] << " ";
    return 0;
}
```

---

4. Reverse an Array using Two Pointers

Swap elements from both ends until met at center

```
#include <bits/stdc++.h>
```

```cpp
using namespace std;

int main()

{

    int n;

    cin >> n;

    int a[n];

    for (int i = 0; i < n; i++)

        cin >> a[i];

    int i = 0, j = n - 1;

    while (i < j)

    {

        swap(a[i], a[j]);

        i++;

        j--;

    }

    for (int i = 0; i < n; i++)

        cout << a[i] << " ";

    return 0;

}
```

---

5. Merge Two Arrays

Combine two arrays into one.

```cpp
#include <bits/stdc++.h>
using namespace std;
```

```cpp
int main()
{
    int n;
    cin >> n;
    int a[n];
    for (int i = 0; i < n; i++)
        cin >> a[i];
    int m;
    cin >> m;
    int b[m];
    for (int i = 0; i < m; i++)
        cin >> b[i];

    int c[n + m];
    for (int i = 0; i < n; i++)
        c[i] = a[i];
    for (int i = 0; i < m; i++)
        c[i + n] = b[i];
    for (int i = 0; i < n + m; i++)
        cout << c[i] << " ";
    return 0;
}
```

---

6. Merge Two Sorted Arrays

Merge two sorted arrays into one sorted array using two-pointers technique.

```cpp
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin >> n;
    int a[n];
    for (int i = 0; i < n; i++)
        cin >> a[i];
    int m;
    cin >> m;
```

```
    int b[m];
    for (int i = 0; i < m; i++)
        cin >> b[i];

    int c[n + m];
    int k = 0;

    int i = 0, j = 0;
    while (i < n && j < m)
    {
        if (a[i] < b[j])
        {
            c[k++] = a[i];
            i++;
        }
        else
        {
            c[k++] = b[j];
            j++;
        }
    }

    while (i < n)
    {
        c[k++] = a[i];
        i++;
    }
    while (j < m)
    {
        c[k++] = b[j];
        j++;
    }

    for (int i = 0; i < n + m; i++)
        cout << c[i] << " ";
    return 0;
}
```

---

7. Check if an Array is Sorted

Compare each element with the next; if any element is greater, it's not sorted.

```cpp
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin >> n;
    int a[n];
    for (int i = 0; i < n; i++)
        cin >> a[i];

    bool ans=true;
    for(int i=0;i<n-1;i++)
    {
        if(a[i]>a[i+1]) {
            ans=false;
            break;
        }
    }
    cout<<ans<<endl;
    return 0;
}
```