|  |  |
| --- | --- |
| **Remove Duplicates in C++** | |
| #include <iostream>  using namespace std;  int removeDup(int arr[], int n) {  int res = 1;  for (int i = 1; i < n; i++) {  if (arr[i] != arr[res - 1]) {  arr[res] = arr[i];  res++;  }  }  return res;  }  int main() {  int arr[] = {2, 2, 3, 4, 5, 6};  int n = sizeof(arr) / sizeof(arr[0]);  int p = removeDup(arr, n);  cout << "After Removal" << endl;  for (int i = 0; i < p; i++) {  cout << arr[i] << " ";  }  cout << endl;  return 0;  } | **Dry Run Table**   | **i** | **arr[i]** | **arr[res - 1]** | **Condition Met (!=)** | **Action** | **arr[res]** | **res** | | --- | --- | --- | --- | --- | --- | --- | | 1 | 2 | 2 | ❌ No | Skip | - | 1 | | 2 | 3 | 2 | ✅ Yes | arr[1] = 3 | 3 | 2 | | 3 | 4 | 3 | ✅ Yes | arr[2] = 4 | 4 | 3 | | 4 | 5 | 4 | ✅ Yes | arr[3] = 5 | 5 | 4 | | 5 | 6 | 5 | ✅ Yes | arr[4] = 6 | 6 | 5 |   **✅ Final Values:**   * res = 5 → means 5 unique elements. * Modified array (first res elements):   arr[] = {2, 3, 4, 5, 6} |
| After Removal  2 3 4 5 6 | |