|  |  |
| --- | --- |
| **Frequency in C++** | |
| #include <iostream>  #include <unordered\_map> // for unordered\_map  using namespace std;  void countFreq(int arr[], int n) {  unordered\_map<int, int> hmp; // Declaration of unordered\_map to store element frequencies    // Count frequencies of each element in the array  for (int i = 0; i < n; i++) {  int key = arr[i];  if (hmp.find(arr[i]) != hmp.end()) {  hmp[arr[i]]++;  } else {  hmp[arr[i]] = 1;  }  }    // Print the frequencies  for (auto itr = hmp.begin(); itr != hmp.end(); itr++) {  cout << itr->first << " " << itr->second << endl;  }  }  int main() {  int arr[] = {4,4,5,2,3,1,6,7,6};  int n = sizeof(arr) / sizeof(arr[0]);  countFreq(arr, n);  return 0;  } | **Dry Run of countFreq(arr, n)**  **Input:**  arr = {4, 4, 5, 2, 3, 1, 6, 7, 6};  n = 9;  **Step 1: Initialize unordered\_map<int, int> hmp**   * hmp is empty at the beginning.   **Step 2: Count Frequencies of Elements**   | **Iteration** | **arr[i]** | **hmp (after processing arr[i])** | | --- | --- | --- | | i = 0 | 4 | {4: 1} | | i = 1 | 4 | {4: 2} | | i = 2 | 5 | {4: 2, 5: 1} | | i = 3 | 2 | {4: 2, 5: 1, 2: 1} | | i = 4 | 3 | {4: 2, 5: 1, 2: 1, 3: 1} | | i = 5 | 1 | {4: 2, 5: 1, 2: 1, 3: 1, 1: 1} | | i = 6 | 6 | {4: 2, 5: 1, 2: 1, 3: 1, 1: 1, 6: 1} | | i = 7 | 7 | {4: 2, 5: 1, 2: 1, 3: 1, 1: 1, 6: 1, 7: 1} | | i = 8 | 6 | {4: 2, 5: 1, 2: 1, 3: 1, 1: 1, 6: 2, 7: 1} |   **Step 3: Print Frequencies**  4 2  5 1  2 1  3 1  1 1  6 2  7 1 |
| Output: 4 2  5 1  2 1  3 1  1 1  6 2  7 1 | |