|  |  |
| --- | --- |
| **Array Pair Divisible by K in C++** | |
| #include <iostream>  #include <vector>  #include <unordered\_map>  using namespace std;  void sol(const vector<int>& arr, int k) {  unordered\_map<int, int> remainderFreqMap;    for (int val : arr) {  int rem = val % k;  remainderFreqMap[rem]++;  }    for (int val : arr) {  int rem = val % k;  if (rem == 0) {  if (remainderFreqMap[rem] % 2 != 0) {  cout << "false" << endl;  return;  }  } else if (2 \* rem == k) {  if (remainderFreqMap[rem] % 2 != 0) {  cout << "false" << endl;  return;  }  } else {  if (remainderFreqMap[rem] != remainderFreqMap[k - rem]) {  cout << "false" << endl;  return;  }  }  }    cout << "true" << endl;  }  int main() {  vector<int> arr = {22, 12, 45, 55, 65, 78, 88, 75};  int k = 7;  sol(arr, k);  return 0;  } | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Dry Run of sol(arr, k)**  arr = {22, 12, 45, 55, 65, 78, 88, 75};  k = 7;  **Step 1: Compute Remainders and Store in remainderFreqMap**  For each element in arr, compute rem = val % k and store it in the map:   | **Value (val)** | **rem = val % 7** | **remainderFreqMap (after insertion)** | | --- | --- | --- | | 22 | 22 % 7 = 1 | {1: 1} | | 12 | 12 % 7 = 5 | {1: 1, 5: 1} | | 45 | 45 % 7 = 3 | {1: 1, 5: 1, 3: 1} | | 55 | 55 % 7 = 6 | {1: 1, 5: 1, 3: 1, 6: 1} | | 65 | 65 % 7 = 2 | {1: 1, 5: 1, 3: 1, 6: 1, 2: 1} | | 78 | 78 % 7 = 1 | {1: 2, 5: 1, 3: 1, 6: 1, 2: 1} | | 88 | 88 % 7 = 4 | {1: 2, 5: 1, 3: 1, 6: 1, 2: 1, 4: 1} | | 75 | 75 % 7 = 5 | {1: 2, 5: 2, 3: 1, 6: 1, 2: 1, 4: 1} |   Final remainderFreqMap:  {1: 2, 5: 2, 3: 1, 6: 1, 2: 1, 4: 1}  **Step 2: Validate Remainder Pairs**  We check the conditions:   * If rem == 0, count should be even (not applicable here). * If 2 \* rem == k, count should be even (not applicable here). * Otherwise, remainderFreqMap[rem] should match remainderFreqMap[k - rem].  | **Value (val)** | **rem = val % 7** | **Condition** | **Check** | | --- | --- | --- | --- | | 22 | 1 | map[1] == map[6] | ❌ 2 != 1 |   Since the condition fails, we print **"false"** and |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |
| Output: false | |