**CẶP SỐ CÓ TỔNG CHIA HẾT CHO K**

Cho mảng arr chứa các số nguyên có độ dài là n, trong đó n là số chẵn và cho số nguyên k, bạn hãy chia mảng trên thành n/2 cặp sao cho tổng của 2 số nguyên trong mỗi cặp đều chia hết cho k. Đầu ra hiển thị true nếu bạn có thể chia mảng thỏa mãn điều kiện trên, hoặc ngược lại, đầu ra hiển thị false.

*Chú ý ở đây là, nếu đã có cặp (x1, x2) thì x1 không được cặp với x3 nữa. Chẳng hạn, (x1, x2) và (x3, x4) được tính là 2 cặp. Còn (x1, x2) và (x2, x3) thì không.*

***Mỗi số chỉ được cặp với một số duy nhất***

**Input**

Dòng đầu tiên là mảng A, không biết trước số lượng phần tử, mỗi phần tử của mảng cách nhau một khoảng trống.

Dòng thứ hai là số nguyên k.

**Output**

In ra màn hình true, false theo kết quả tương ứng.

**Giới hạn**

* Độ dài mảng không quá 1e5, và là số chẵn
* Các phần tử trong mảng không quá 1e9
* 1 <= k <=1e5

**Ví dụ**

|  |  |
| --- | --- |
| Input | Output |
| 1 2 3 4 5 10 6 7 8 9  5 | true |
| 1 2 3 4 5 6  7 | true |
| 1 2 3 4 5 6  10 | false |

**Giải thích test:** Chỉ số bắt đầu từ 0

Test 1: Có 5 cặp là: (a0, a9); (a1, a8), (a2, a7), (a3, a6), (a4, a5)

**Lời giải**

#include <bits/stdc++.h>

using namespace std;

int main() {

    vector<int> a;

    string s;

    getline(cin, s);

    stringstream ss(s);

    string token;

    while (ss >> token) a.push\_back(stoi(token));

    int n = a.size();

    int k;

    cin >> k;

    unordered\_map<int, int> mp;

    // Fill the remainder map

    for (int i : a) {

        int x = i % k;

        if (x < 0) x += k;

        mp[x]++;

    }

    bool canFormPairs = true;

    for (const auto& entry : mp) {

        int remainder = entry.first;

        int count = entry.second;

        if (remainder == 0) {

            if (count % 2 != 0) {

                canFormPairs = false;

                break;

            }

        } else {

            int complement = k - remainder;

            if (mp[remainder] != mp[complement]) {

                canFormPairs = false;

                break;

            }

        }

    }

    if (canFormPairs) cout << "true";

    else cout << "false";

}