#56115158 | Daniel_Chura's solution for [UVA-10664]

Status	Length	Lang	Submitted	Open	Share text	RemoteRunId
Accepted	1406	C++11 5.3.0	2024-11-17 10:37:32	☑	☐	29969585

```cpp
#include <iostream>
#include <vector>
#include <string>
#include <sstream>
#include <unordered_map>
#include <algorithm>

using namespace std;

unordered_map<string, bool> memo;

string makeKey(int index, int target) {
    return to_string(index) + "," + to_string(target);
}

bool canPartitionMemo(const vector<int>& nums, int index, int target) {
    if (target == 0) return true;
    if (index < 0 || target < 0) return false;

    string key = makeKey(index, target);
    if (memo.find(key) != memo.end()) {
        return memo[key];
    }

    bool include = canPartitionMemo(nums, index - 1, target - nums[index]);
    bool exclude = canPartitionMemo(nums, index - 1, target);

    memo[key] = include || exclude;
    return memo[key];
}

int main() {
    int testCases;
    cin >> testCases;
    cin.ignore();

    while (testCases--) {
        string line;
        getline(cin, line);
        stringstream ss(line);
        vector<int> nums;
        int num, sum = 0;

        while (ss >> num) {
            nums.push_back(num);
            sum += num;
        }

        if (sum % 2 != 0) {
            cout << "NO" << endl;
            continue;
        }

        int target = sum / 2;
        memo.clear();
        if (canPartitionMemo(nums, nums.size() - 1, target)) {
            cout << "YES" << endl;
        } else {
            cout << "NO" << endl;
```