**★★★☆☆**

**題組：Problem Set Archive** **with Online Judge**

**題號：10664: Luggage**

**解題者：賴琮翰**

**解題日期：2021年5月20日**

**題意：**

輸入m組測資，每組測資包含n個(1<=n<=20)行李箱的重量(重量為整數)。

現在有兩台車來載行李，判斷是否能找到某種分法，讓兩車所載行李箱總重量相同。

**題意範例：**

[Sample Input]

3 🡺 3組測資

1 2 1 2 1

2 3 4 1 2 5 10 50 3 50

3 5 2 7 1 7 5 2 8 9 1 25 15 8 3 1 38 45 8 1

[Sample Output]

NO

YES

YES

**解法：**

* 1. 先判斷總重是否為偶數，如果不是偶數則輸出NO。
	2. 把資料sort後，用遞迴的方法討論每個行李箱要不要被選擇，選擇過程中刪除超出範圍的路徑。
	3. Recursive Case
		1. 當前element大於目標 🡺 (不選element，目標不變)
		2. 當前element小於目標

 🡺 (選element，目標減去element的值)

 || (不選element，目標不變)

* 1. Base Case
		1. 當前element等於目標 🡺 return true
		2. 已經找過所有element 🡺 return false
1. 找到可行解輸出YES，反之輸出NO。

**解法範例：**

1 2 2 3 3 4 5 10 50 50

//用branch and bound tree來表達

****

//此題return true，輸出YES

**討論：**

此題由於測資較小，若改使用dp法會比較快。

**程式：**

#include <iostream>

#include <algorithm>

#include <vector>

#include <string>

#include <cctype>

using namespace std;

bool SOS(vector<int>,int,int);

int main()

{

 int testNum;

 string input;

 cin >> testNum;

 cin.ignore();

 for(int i=0;i<testNum;i++)

 {

 vector<int> subset;

 int total=0;

 bool flag=false;

 getline(cin,input);

 for(int j=0;j<input.size();j++)

 {

 int temp=0;

 if(isdigit(input[j]))

 {

 while(isdigit(input[j]))

 {

 temp\*=10;

 temp+=input[j]-'0';

 j++;

 }

 subset.push\_back(temp);

 }

 }

 for(int j=0;j<subset.size();j++)

 total+=subset[j];

 if(total%2!=0)

 cout << "NO" << endl;

 else

 {

 total=total/2;

 sort(subset.begin(),subset.end());

 if(SOS(subset,subset.size(),total))

 cout << "YES" << endl;

 else

 cout << "NO" << endl;

 }

 }

}

bool SOS(vector<int>set,int n,int sum)

{

 if(sum==0)

 return true;

 if(n==0&&sum!=0)

 return false;

 if(set[n-1]>sum)

 return SOS(set,n-1,sum);

 return SOS(set,n-1,sum)||SOS(set,n-1,sum-set[n-1]);

}