#include <iostream>

using namespace std;

bool isSubsetSum(int set[],int n, int sum){

bool subset[n+1][sum+1];

for(int i=0;i<=n;i++){

subset[i][0]=true;

}

for(int i=1;i<=sum;i++){

subset[0][i]=false;

}

for(int i=1;i<=n;i++){

for(int j=1;j<=sum;j++){

if(set[i-1]>j){

subset[i][j]=subset[i-1][j];

}

if(set[i-1]<=j){

subset[i][j]= subset[i-1][j] || subset[i-1][j-set[i-1]] ;

}

}

}

return subset[n][sum];

}

bool findPartiion(int arr[],int n){

int sum=0;

for(int i=0;i<n;i++){

sum+=arr[i];

}

if(sum%2!=0){

return false;

}else{

return isSubsetSum(arr,n,sum/2);

}

}

int main() {

int arr[] = {3, 1, 1, 2, 2, 1};

int n = sizeof(arr) / sizeof(arr[0]);

if (findPartiion(arr, n) == true)

cout << "Can be divided into two subsets of equal sum";

else

cout << "Can not be divided into"

<< " two subsets of equal sum";

return 0;

}