**Partition Equal Subset Sum**

**Medium**

Given an array **arr[]** of size **N**, check if it can be partitioned into two parts such that the sum of elements in both parts is the same.

**Example 1:**

**Input:** N = 4

arr = {1, 5, 11, 5}

**Output:** YES

**Explanation:**

The two parts are {1, 5, 5} and {11}.

**Example 2:**

**Input:** N = 3

arr = {1, 3, 5}

**Output:** NO

**Explanation:** This array can never be

partitioned into two such parts.

**Expected Time Complexity:** O(N\*sum of elements)  
**Expected Auxiliary Space:** O(N\*sum of elements)

**Constraints:**  
1 ≤ N ≤ 100  
1 ≤ arr[i] ≤ 1000  
N\*sum of elements ≤ 5\*106

**Company Tags**

[**Accolite**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Accolite) [**Amazon**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Amazon) [**Microsoft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Microsoft) [**OYO Rooms**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=OYO%20Rooms) [**Adobe**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Adobe) [**Drishti-Soft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Drishti-Soft)

**Topic Tags**

[**Dynamic Programming**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Dynamic%20Programming) [**subset**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=subset) [**Algorithms**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Algorithms)

//{ Driver Code Starts

// Initial Template for Java

import java.io.\*;

import java.util.\*;

class CodingMaxima{

public static void main(String args[])throws IOException

{

BufferedReader in = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(in.readLine());

while(t-- > 0){

int N = Integer.parseInt(in.readLine());

String input\_line[] = in.readLine().trim().split("\\s+");

int arr[] = new int[N];

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

arr[i] = Integer.parseInt(input\_line[i]);

Solution ob = new Solution();

int x = ob.equalPartition(N, arr);

if(x == 1)

System.out.println("YES");

else

System.out.println("NO");

}

}

}

// } Driver Code Ends

// User function Template for Java

class Solution{

static int equalPartition(int N, int arr[])

{

int sum=0;

for(int num: arr)

sum+=num;

if(sum%2!=0)

return 0;

int[] dp=new int[N+1];

return helper(0, N, arr, sum/2, dp);

}

private static int helper(int i, int n, int[] arr, int target, int[] dp){

if(target==0)

return 1;

if(i==n)

return 0;

if(dp[i]!=0){

return dp[i];

}

int with=0;

if(target>=arr[i])

with=helper(i+1, n, arr, target-arr[i], dp);

int without=helper(i+1, n, arr, target, dp);

return dp[i]=with | without;

}

}