Narmadha S
240801216
ECE D

QuestionText1

SunnyandJohnnyliketopooltheirmoneyandgototheice cream parlor. Johnny never buys the same flavor that Sunnydoes.Theonlyotherruletheyhaveisthattheyspendall of their money.
Givenalistofpricesfortheflavorsoficecream,selectthetwo that will cost all of the money they have.
Forexample,theyhave*m=6*tospendandtherearеflavors costing *cost = [1, 2, 3, 4, 5, 6]*. The two flavors costing *1*and*5*meetthecriteria.Using*1*-basedindexing,theyareat indices *1* and *4*.
FunctionDescription
Completethecodeintheeditorbelow.Itshouldreturnanarray containing the indices of the prices of the two flavors theybuy.
Ithasthefollowing:
· m:anintegerdenotingtheamountofmoneytheyhaveto spend
· cost:anintegerarraydenotingthecostofeachflavorofice cream
InputFormat
Thefirstlinecontainsaninteger,*t*,denotingthenumberoftrips to the ice cream parlor. The next *t* sets of lines each describeavisit.Eachtripisdescribedasfollows:
1. Theinteger*m*,theamountofmoneytheyhavepooled.
2. Theinteger*n*,thenumberofflavorsofferedatthetime.

Edit with WPS Office

3. *n*space-separatedintegersdenotingthecostofeachflavor: *cost[cost[1],cost[2],...,cost[n]]*.

Note: Theindexwithinthecostarrayrepresentstheflavorof the ice cream purchased.

Constraints

68· *1≤t≤50*

· *2≤m≤104*

· *2≤n≤104*

· *1≤cost[i]≤104,"iÎ[1,n]*

·Therewillalwaysbeauniquesolution.

OutputFormat

For each test case, print two space-separated integers denotingtheindicesofthetwoflavorspurchased,inascending order.

SampleInput

2

4

5

14532

4

4

2243

SampleOutput

14

12

Explanation

SunnyandJohnnymakethefollowingtwotripstotheparlor:

1. Thefirsttime,theypooltogether*m=4*dollars.Ofthefive flavors available that day, flavors *1* and *4* have a total costof*1+3=4*.

2. Thesecondtime,theypooltogether*m=4*dollars.TOfthe four flavors available that day, flavors *1* and *2* have a totalcostof*2+2=4*.

Edit with WPS Office

```c
#include<stdio.h>
int main(){
int t,m,n,c=0;
scanf("%d",&t);
for(int i=0;i<t;i++)
{
    c=0;
    scanf("%d\n%d",&m,&n);
    int arr[n];
    for(int j=0;j<n;j++)
    {
        scanf("%d",&arr[j]);
    }
    for(int a=0;a<n-1;a++)
    {
        for(int b=a+1;b<n;b++)
        {
            if(arr[a]+arr[b]==m)
            {
                printf("%d %d\n",a+1,b+1);
                c=1;break;

            }
        }
       if(c==1)
        {
            break;
        }
    }
}
}
```

	Input	Expected	Got	
✓	2 4 5 1 4 5 3 2 4 4 2 2 4 3	1 4 1 2	1 4 1 2	✓

Passed all tests! ✓

Questiontext2

NumerostheArtisthadtwolistsThatwerepermutationsofone another. He was very proud. Unfortunately, while transporting them from one exhibition to another, some numberswerelostoutofthefirstlist.Canyoufindthemissing numbers?

Asanexample,thearraywithsomenumbersmissing,***arr=[7,2, 5, 3, 5, 3]***. The original array of numbers ***brr = [7, 2,5,4,6,3,5,3]***.Thenumbersmissingare***[4,6]***.

Notes

· Ifanumberoccursmultipletimesinthelists,youmustensure that the frequency of that number in both lists is thesame.Ifthatisnotthecase,thenitisalsoamissing number.

.Youhavetoprintallthemissingnumbersinascendingorder.

Edit with WPS Office

· Printeachmissingnumberonce,evenifitismissingmultiple times.
· Thedifferencebetweenmaximumandminimumnumberin the second list is less than or equal to *100*.
Completethecodeintheeditorbelow.Itshouldreturnanarray of missing numbers.
Ithasthefollowing:
· arr:thearraywithmissingnumbers
· brr:theoriginalarrayofnumbers

InputFormat

Therewillbefourlinesofinput:
n-thesizeofthefirstlist,*arr*
Thenextlinecontains*n*space-separatedintegers*arr[i]m* - the size of the second list, *brr*
Thenextlinecontains*m*space-separated integers*brr[i]*

Constraints

· *1≤n,m≤2x105*
· *n≤m*
· *1≤brr[i]≤2x104*
· *Xmax−Xmin<101*

OutputFormat

Outputthemissingnumbersinascendingorder.

SampleInput

10
203204205206207208203204205206
13
203204204205206207205208203206205206204

SampleOutput

204205206

Explanation

*204*ispresentinbotharrays.Itsfrequencyin*arr*is*2*,whileits frequency in *brr* is *3*. Similarly, *205* and *206* occur

Edit with WPS Office

twicein*arr*,butthreetimesin*brr*.Therestofthenumbers have the same frequencies in both lists.

```c
#include<stdio.h>
int main(){
    int n,m,c1=0,co,c;
    scanf("%d",&n);
    int arr[n];
    for(int a=0;a<n;a++){
        scanf("%d",&arr[a]);
    }
    scanf("%d",&m);
    int brr[m];
    int ans[m];
    for(int b=0;b<m;b++){
        scanf("%d",&brr[b]);
    }
    for(int j=0;j<m;j++)
    {
        c=0;
        for(int i=0;i<n;i++){
            if(arr[i]==brr[j]){
                c=1;
                arr[i]=-1;
                break;
            }
    }
    if(c==0)
    {
    ans[c1]=brr[j];
    c1++;
    }
    }
    for(int a=0;a<c1;a++)
    {
        co=0;
        for(int b=0;b<c1;b++){
            if(ans[b]<ans[a])
            co++;
        }
        int temp=ans[a];
        ans[a]=ans[co];
        ans[co]=temp;
    }
    for(int i=0;i<c1;i++)
    printf("%d ",ans[i]);
    return 0;
    }
```

	Input	Expected	Got	
✓	10 203 204 205 206 207 208 203 204 205 206 13 203 204 204 205 206 207 205 208 203 206 205 206 204	204 205 206	204 205 206	✓

Passed all tests! ✓

Questiontext3

WatsongivesSherlockanarrayofintegers.Hischallengeisto find an element of the array such that the sum of all elementstotheleftisequaltothesumofallelementstothe right. For instance, given the array *arr = [5, 6, 8, 11]*, *8* isbetweentwosubarraysthatsumto*11*.Ifyourstartingarray is *[1]*, that element satisfies the rule as left and right sumto*0*.

Edit with WPS Office

Youwillbegivenarraysofintegersandmustdetermine whether there is an element that meets the criterion.
Completethecodeintheeditorbelow.Itshouldreturnastring, either YES if there is an element meeting the criterion orNOotherwise.
Ithasthefollowing:
· arr:anarrayofintegers

InputFormat

The first line contains *T*, the number of test cases.
Thenext *T*pairsoflineseachrepresentatestcase.
- Thefirstlinecontains*n*,thenumberofelementsinthearray *arr*.
- Thesecondlinecontains*n*space-separatedintegers*arr[i]* where*0≤i<n*.

Constraints

· *1≤T≤10*
· *1≤n≤105*
· *1≤arr[i]≤2x104*
· *0≤i≤n*

OutputFormat

ForeachtestcaseprintYESifthereexistsanelementinthe array, such that the sum of the elements on its left is equalto the sum of the elements on its right; otherwise print NO.

Sample Input 0

2
3
123
4
1233

SampleOutput0

NO
YES

Explanation0

Edit with WPS Office

Forthefirsttestcase,nosuchindexexists.
Forthesecondtestcase,***arr[0]+arr[1]=arr[3]***,therefore index *2* satisfies the given conditions.

SampleInput1

733
5
11411
4
2000
4
0020

SampleOutput1

YES
YES
YES

Explanation1

Inthefirsttestcase,***arr[2]=4***isbetweentwosubarrays summing to *2*.
Inthesecondcase,***arr[0]=2***isbetweentwosubarrays summing to *0*.
Inthethirdcase,***arr[2]=2***isbetweentwosubarrayssumming to *0*.

Edit with WPS Office

```c
#include<stdio.h>
int main()
{
    int t,n,is,rs,m;
    scanf("%d",&t);
    for(int i=0;i<t;i++){
        is=0;
        rs=0;
        scanf("%d",&n);
        int arr[n];
        for(int j=0;j<n;j++)
        scanf("%d",&arr[j]);
        m=n/2;
        if(arr[m]==0){
            for(m=0;arr[m]==0&&m<n;m++);
        }
        for(int j=0;j<=m;j++)
        is=is+arr[j];
        for(int j=m;j<n;j++)
        rs=rs+arr[j];
        printf("%s\n",(is==rs)?"YES":"NO");
    }
}
```

	Input	Expected	Got	
✓	3 5 1 1 4 1 1 4 2 0 0 0 4 0 0 2 0	YES YES YES	YES YES YES	✓
✓	2 3 1 2 3 4 1 2 3 3	NO YES	NO YES	✓

Passed all tests! ✓

Edit with WPS Office