Searching Algorithms – Linear and Binary

Ex.No.:   **Date:**

IceCreamParlor

ProblemStatement:

Sunny and Johnny like to pool their money and go to the ice cream parlor. Johnny never buys the same flavor that Sunny does. The only other rule they have is that they spend all of their money.

Given a list of prices for the flavors of ice cream, select the two that will cost all of the money they have.

Forexample,theyhavem=6tospendandthereareflavorscostingcost=[1,2,3, 4,5,6].Thetwoflavorscosting1and5meetthecriteria.Using1-basedindexing,theyare   at indices 1 and 4.

Complete the code in the editor below. It should return an array containing the indices of the prices of thetwoflavors they buy, sortedascending.
Ithasthefollowing:

- m:anintegerdenotingtheamountofmoneytheyhavetospend
- cost:anintegerarraydenotingthecostofeachflavoroficecream

InputFormat
Thefirstlinecontainsaninteger,t,denotingthenumberoftripstotheicecreamparlor. The next t sets of lines each describe a visit. Each trip is described as follows:

1. Theintegerm,theamountofmoneytheyhavepooled.
2. Theintegern,thenumberofflavorsofferedatthetime.
3. nspace-separatedintegersdenotingthecostofeachflavor:cost[cost[1],cost[2],. .., cost[n]].

Note:Theindexwithinthecostarrayrepresentstheflavoroftheicecreampurchased.

Constraints

- 1≤t≤50
- 2≤m≤104
- 2≤n≤104
- 1≤cost[i]≤104,∀i∈[1,n]
- Therewillalwaysbeauniquesolution.

OutputFormat
For eachtest case, print two space-separated integers denoting the indices of the two flavors purchased, in ascending order.

SampleInput
2
4
5
14532
4
4
2243

SampleOutput14
12

Program:

Answer: (penalty regime: 0 %)

```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! ✓

Ex.No.:   **Date:**

MissingNumbers

ProblemStatement:

NumerostheArtisthadtwoliststhatwerepermutationsofoneanother.Hewasveryproud. Unfortunately, while transporting them from one exhibition to another, some numbers were lost out of the first list. Can you find the missing numbers?

Asanexample,thearraywithsomenumbersmissing,arr=[7,2,5,3,5,3].Theoriginal arrayofnumbersbrr=[7,2,5,4,6,3,5,3].Thenumbersmissingare[4,6].

Notes
- If a number occurs multiple times in the lists, you must ensure that the frequency of that number in both lists is the same. If that is not the case, then it is also a missing number.
- Youhavetoprintallthemissingnumbersinascendingorder.
- Printeachmissingnumberonce,evenifitismissingmultipletimes.
- The difference between maximum and minimum number in the second list is less than or equal to 100.

Complete the code in the editor below. It should return a sorted array of missing numbers. It has the following:
- arr:thearraywithmissingnumbers
- brr:theoriginalarrayofnumbers

InputFormat
Therewillbefourlinesofinput:
n-thesizeofthefirstlist,arr
The next line contains n space-separated integers arr[i]
m - the size of the second list, brr
Thenext line containsmspace-separatedintegers brr[i]

Constraints
        $1 \le n, m \le 2 \times 10^5, n \le m, 1 \le brr[i] \le 2 \times 10^4, Xmax - Xmin < 101$

OutputFormat
Outputthemissingnumbersinascendingorder.

SampleInput
10
203204205206207208203204205206
13
203204204205206207205208203206205206204

SampleOutput
204205206

Program:

```c
#include<stdio.h>
int main()
{
    int n,m,c,c1=0,co;
    scanf("%d",&n);
    int arr[n];
    for(int a=0;a<n;a++){
        scanf("%d",&arr[a]);
    }
    scanf("%d",&m);
    int brr[m],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
✓	10 203 204 205 206 207 208 203 204 205 206 13 203 204 204 205 206 207 205 208 203 206 205

Passed all tests! ✓

Ex.No.:   **Date:**

SherlockandArray

ProblemStatement:

Watson gives Sherlock an array of integers. His challenge is to find an element ofthe array such that the sum of all elements to the left is equal to the sum of all elements to the right. For instance, given the array arr = [5, 6, 8, 11], 8 is between two subarrays that sum to 11. If your starting array is [1], that element satisfies the rule as left and right sumto 0. You will be given arrays of integers and must determine whether there is an element that meets the criterion.

Completethecodeintheeditorbelow.Itshouldreturnastring,eitherYESifthere isanelementmeetingthecriterionorNOotherwise.Ithasthefollowing:arr:anarrayof integers

InputFormat
The first line contains T, the number of test cases.
The next T pairs of lines each represent a test case.
- Thefirstlinecontainsn,thenumberofelementsinthearrayarr.
- Thesecondlinecontainsnspace-separatedintegersarr[i]where0≤i<n.

Constraints: $1 \le T \le 10, 1 \le n \le 105$, $1 \le arr[i] \le 2 \times 104$, $0 \le i \le n$

OutputFormat
For each test case print YES if there exists an element in the array, such that the sum of the elements on its left is equal to the sum of the elements on its right; otherwise print NO.

SampleInput0
2
3
123
4
1233

SampleOutput0
NO
YES

Program:

```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! ✓