Qus 37: Take as input N, the size of array. Take N more inputs and store that in an array. Take as input "target", a number. Write a function which prints all pairs of numbers which sum to target.

Input Format

The first line contains input N. Next N lines contains the elements of array and (N+1)th line contains target number.

Constraints

Length of the arrays should be between 1 and 1000.

Output Format

Print all the pairs of numbers which sum to target. Print each pair in increasing order.

Sample Input

5

1

3

4

2

5

5

Sample Output

1 and 4

2 and 3

Explanation

Find any pair of elements in the array which has sum equal to target element and print them.

```
void checkPair(int arr[],int n,int target)
{
    for(int i=0;i<n;i++)
    {
        for(int j=i+1;j<n;j++)
        if(arr[i]+arr[j]==target)
        {
            cout<<arr[i]<<" and "<<arr[j]<<endl;
        }
    }
}
```

Qus 38- https://www.geeksforgeeks.org/segregate-0s-and-1s-in-an-array-by-traversing-array-once/

```
    void binS1ort(int A[], int N)
    {

        int l=0,h=N-1;
        while(l<h)
        {
            while(A[l]==0&&l<h)l++;
            while(A[h]==1&&l<h)h--;
            if(l<h)
            {

                A[l]=0;
            A[h]=1;
            l++;h--;
        }
    }
}
```

Qus 39- Take as input N, the size of array. Take N more inputs and store that in an array. Take as input “target”, a number. Write a function which prints all triplets of numbers which sum to target.

Input Format

First line contains input N.

Next line contains N space separated integers denoting the elements of the array.

The third line contains a single integer T denoting the target element.

Constraints

Length of Array should be between 1 and 1000.

Output Format

Print all the triplet present in the array in a new line each. The triplets must be printed as A, B and C where A,B and C are the elements of the triplet (A<=B<=C) and all triplets must be printed in sorted order. Print only unique triplets.

Sample Input

```
9
5 7 9 1 2 4 6 8 3
10
```

Sample Output

1, 2 and 7

1, 3 and 6

1, 4 and 5

2, 3 and 5

Explanation

Array = {5, 7, 9, 1, 2, 4, 6 ,8 ,3}. Target number = 10. Find any three number in the given array which sum to target number.

```
void printTriplet(int arr[],int n,int target)
{
   sort(arr,arr+n);
        int l,h;
        for(int i=0;i<n-2;i++)
        {

                int l=i+1;
                int h=n-1;
                while(l<h)
                {

                        if(arr[i]+arr[l]+arr[h]==target)
                        {

                                cout<<arr[i]<<" "<<arr[l]<<" "<<arr[h]<<endl;
                                l++;h--;
                        }
                        else if(arr[i]+arr[l]+arr[h]>target)
                        h--;
```

```
            else l++;
        }
    }
}
```

Qus 40- https://www.techiedelight.com/trapping-rain-water-within-given-set-bars/

```
int trappingWater(int arr[], int n){
    // Code here
    int ml[n],mr[n];
    ml[0]=arr[0];
    mr[n-1]=arr[n-1];
    for(int i=1;i<n;i++)
    {
      ml[i]=max(ml[i-1],arr[i]);

    }
    for(int j=n-2;j>=0;j--)
    mr[j]=max(mr[j+1],arr[j]);

    int temp[n],sum=0;
    for(int i=0;i<n;i++)
    {
      sum+=min(ml[i],mr[i])-arr[i];
    }return sum;
  }
```

Qus 41- https://www.geeksforgeeks.org/largest-sum-contiguous-subarray/

```
int maxSubarraySum(int arr[], int n){

        // Your code here
        int total_sum=arr[0];
        int temp=arr[0];
        for(int i=1;i<n;i++)
        {
            temp=max(arr[i],arr[i]+temp);
            total_sum=max(total_sum,temp);
        }
        return total_sum;



    }
```

Qus 42- Take as input N, the size of array. Take N more inputs and store that in an array. Take as input M, the size of second array and take M more inputs and store that in second array. Write a function that returns the sum of two arrays. Print the value returned.

Input Format

Constraints

Length of Array should be between 1 and 1000.

Output Format

Sample Input

```
4

1 0 2 9

5

3 4 5 6 7
```

Sample Output

```
3, 5, 5, 9, 6, END
```

Explanation

Sum of [1, 0, 2, 9] and [3, 4, 5, 6, 7] is [3, 5, 5, 9, 6] and the first digit represents carry over , if any (0 here) .

```
#include<bits/stdc++.h>

using namespace std;



int main()

{


    int n,m;

    cin>>n;

    int a[n];

    for(int i=0;i<n;i++)cin>>a[i];

    cin>>m;int b[m];

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

    {


        cin>>b[i];

    }int l;

    if(n<m)
```

```
{
    int res[m];
    int carry=0;
    int sum=0;
    int i=m-1,j=n-1;l=m-1;
    while(j>=0)
    {
        sum=a[j--]+b[i--]+carry;

        res[l--]=sum%10;
        carry=sum/10;

    }
    while(i>=0)
    {

        sum=carry+b[i--];
        res[l--]=sum%10;
        carry=sum/10;
    }for(int i=0;i<m;i++)
    cout<<res[i]<<" ";
}
else{

    int res[n];
    int carry=0;
    int sum=0;
```

```
        int i=m-1,j=n-1; l=n-1;
        while(i>=0)
        {
            sum=a[j--]+b[i--]+carry;

            res[l--]=sum%10;
            carry=sum/10;

        }
        while(j>=0)
        {

            sum=carry+a[j--];
            res[l--]=sum%10;
            carry=sum/10;
        }for(int i=0;i<n;i++)
        cout<<res[i]<<" ";

    }

}
```

Qus 43- https://leetcode.com/problems/third-maximum-number/

```
int thirdMax(vector<int>& nums) {
    sort(nums.begin(),nums.end());
    set<int>s;
     int n=nums.size();
    if(n<3)
    {
        return nums[n-1];
    }else{

        set<int>s;
        for(int i=n-1;i>=0;i--)
        {

                s.insert(nums[i]);

              if(s.find(nums[i])!=s.end()&&s.size()==3)
            {
                return nums[i];
            }

        }
    }

    return nums[n-1];
}
```

Qus 44- https://leetcode.com/problems/valid-triangle-number/

Qus 45- https://www.tutorialspoint.com/counting-frequencies-of-array-elements-in-cplusplus

Qus 46- Insertion Sort

```
void insertionSort(int arr[],int n)
{

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

                int j=i-1;
                int temp=arr[i];
                while(arr[j]>temp&&j>=0)
                {

                        arr[j+1]=arr[j];
                        j--;
                }
                arr[j+1]=temp;
        }
        print(arr,n);
}
```

Communication Qus ----- Explain something about yourself which is not mentoined in the Resume.