COMPETITIVE CODING
with
RAMAN CLASSES

... An interactive series for those willing to learn and code

Problem : Program to find a pair of an array, whose sum is X

Topic	: **Arrays**
Difficulty	: **EASY**
Programming Language	: **C++**
Time to Spend	: **15 min**

Problem Statement

Write a program in C++ to find a pair of elements in the given array such that its sum is X. If not present print -1.

Input :

6 8
3 7 4 5 9 2

Output :

(3,5)

Input Description :

The first line contains two integers ***n*** and ***X*** denoting the number of elements in the array and the sum.
The next line contains ***n*** space-separated integers denoting the given array.

Let Us Revise

In order to solve this problem, go through the following concepts.

1. Sets (more precisely, unordered sets)
2. Loop statements.

Problem Description

3	7	4	5	9	2

Given an array of integers, we need to find **a** pair of elements such that its sum is equal to X.
Assuming X to be 8.

The pair is

(3,5)

If there is no such pair present in the array, print -1
i.e. Assuming X to be 20.

There is no such pair having sum equal to 20, that's why -1 will be the output.

Let Us Think

In order to solve this problem, let us think and analyse how to get started with this problem.

Try to think how can we say that a particular element(Say A) of the array is one of the elements of the pair.

After giving some thought to it, we can see if we can find an element having value X-A then we can say there exist a pair (A, X-A), whose sum is X.

Let Us Think

So the problem boils down to finding if there is any element present having value (X-A) for any element A, excluding A.

Let Us Code

Now you know the logic of code . Lets proceed with the coding part.

Let Us Code

Now you know the logic of code . Lets proceed with the coding part.

Things we need to do in this problem :-

1. **Maintain a set to keep track of all elements of array iterated so far.**
2. **A function to check if there is any element present in the Set, having value (X - Value of current element of array).**

3. **A code to print -1 in case no such pair is found.**

1. **Maintain a set to keep track of all elements of array iterated so far.**

Assuming the array is ‘a’ of size n and Unordered Set as s.

```
//Maintaining an Unordered Set of elements of Array.
for(int i=0;i<n;i++){

    int complement=X-a[i];        // X-A

    //Checking whether X-A is present or not and also updating flag value if
                                    check function is true.
    if(check(complement, s)){
        cout<<"("<<complement<<","<<a[i]<<")"<<endl;
        flag=1;
        break;
    }

    s.insert(a[i]);
}
```

2. A function to check if there is any element present in the Set, having value (X - Value of current element of array).

```
//Function to check whether X-A is present in the set or not.

bool check(int complement, unordered_set<int> s){
    if(s.find(complement) != s.end()){
        return true;
    }
    return false;
}
```

3. A code to print -1 in case no such pair is found.

```
//code to print -1 in case no pair is found
if(flag==0){
    cout<<"-1"<<endl;
}
return 0;
```

Tweak n Try

Try to solve the problem in O(1) Space Complexity.

Hint : Sorting technique And Two pointer Technique

Note: The Solution will be of O(nlogn) complexity i.e. inefficient than the shown solution but is fun to solve and very important in case its provided that array is always sorted.
So, do give it a try.

Thank You !

... brought to you by

www.ramanclasses.in