COMPETITIVE CODING
with
RAMAN CLASSES

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

Problem : Magical Treasure

Topic	: **Arrays**
Difficulty	: **EASY**
Programming Language	: **C++**
Time to Spend	: **10 min**

Problem Statement

Problem:
Our friends Alpha and Beta found a magical treasure of Asgard. It consists of piles of gold coins. It is magical since if anyone tries to take a pile of coins, all the other piles of **exactly** coins (if they exist) disappear.

Alpha and Beta only have **one turn each** to choose a pile for themselves starting with Alpha. In one turn, a complete pile of gold coins can be chosen and since our friends are smart they will choose the pile with the maximum coins.

Find the number of coins Beta will get in his turn.

Problem Statement

Input :

6

1 2 3 2 1 3

Output :

2

Input Description :

First line of the input contains a single integer , number of piles.

Second line of the input contains space seperated integers, number of gold coins in each pile.

Problem Statement

Output Description :

Single integer which is the number of coins Beta will receive in his first turn.

Let Us Revise

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

1. Using Loop Statements

Problem Description

We are given a array, whose elements represent number of coins in that pile and Alpha's turn is first. So **according to question**, alpha will select the pile having maximum gold coin and same number of coins will be deleted from every other pile if present.

And we need to output, what will be maximum number of coins that will be available for Beta in his turn, in a pile.

For example: if array= **[2,3,1,2,1,3**] , Alpha will choose the maximum element i.e. 3 and delete same number of coins if present from all other piles.
The resultant array for Beta's turn will be **[2,1,2,1]**
Thus the maximum element now is 2. So,
Output : 2

Let Us Think

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

Try to solve it on paper using your own example. On careful observation, you can see the number of gold coin available to beta is second largest pile of gold coins.

Now the problem boils down to finding the second largest number in the array given.

Let Us Think

But how we actually do that?

Let Us Think

But how we actually do that?

Maintain a variable(say a) to store the largest value of the array.
And
Another variable(say b) to store the second largest value of the array.

1. If the current element is > a then,
b=a and a= current element.
2. If the current element is > a but < b then,
b= current element.

Let Us Think

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

Let Us Think

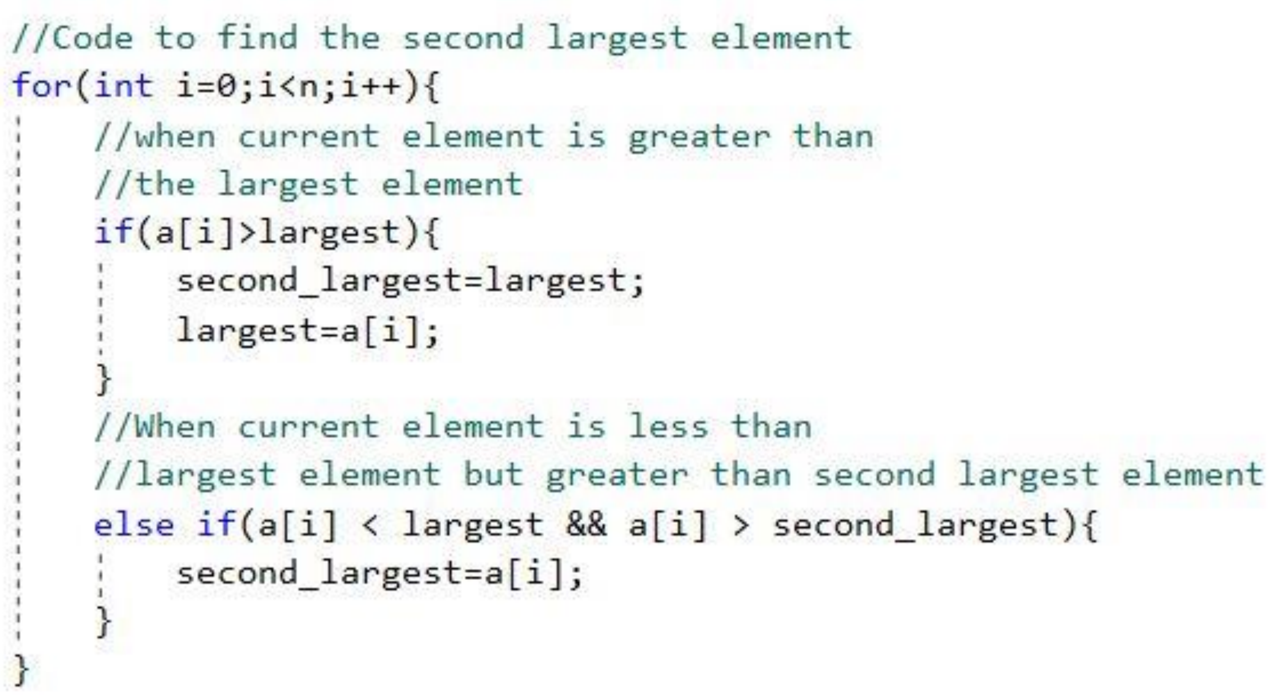
Now you know the logic of the code lets proceed with the coding part.

Things we need to do in this problem:

Find the second largest number in the given array.

Let Us Code

Find the second largest number in the given array.



```
//Code to find the second largest element
for(int i=0;i<n;i++){
    //when current element is greater than
    //the largest element
    if(a[i]>largest){
        second_largest=largest;
        largest=a[i];
    }
    //When current element is less than
    //largest element but greater than second largest element
    else if(a[i] < largest && a[i] > second_largest){
        second_largest=a[i];
    }
}
```

Pitfall Prevention

Our logic is ready. But million dollar question is – ***IS IT CORRECT ?***

Yes it is.

Not given in question.

But, what if there is no second largest number ?
For example: array=[1,1,1,1,1,1].

We should check whether second is equal to its initialized value,
If yes, then there is no second largest element.
If no, then there is a second largest element.

Thank You !

... brought to you by

www.ramanclasses.in