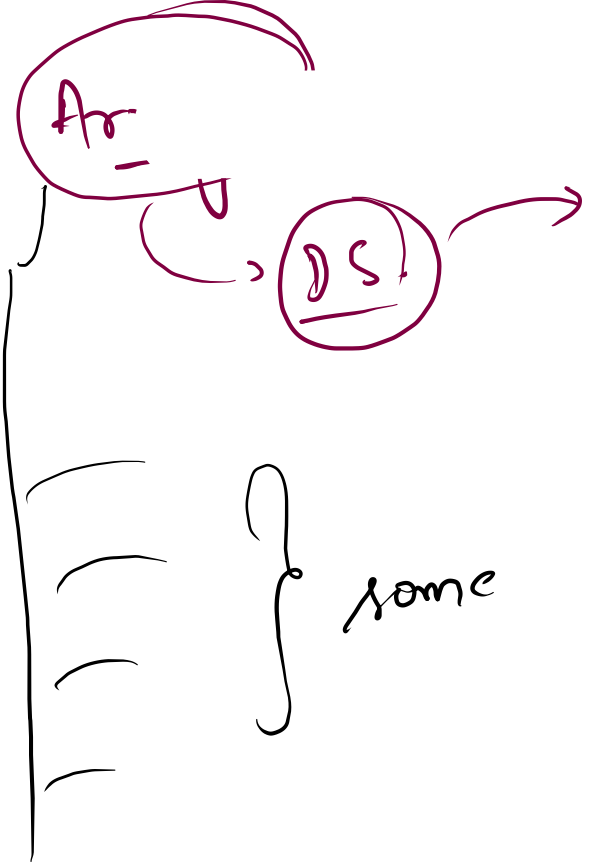

Arr
DS
similar data.
Nested loops.
some

Greater Than Me

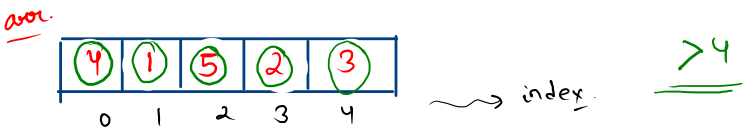
Given an array then for each index print the count of the elements which are strictly greater than the element present at that index.

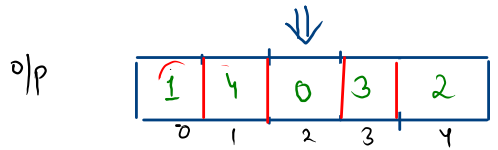
Sample Input 0

```
5
1 2 3 4 5
```

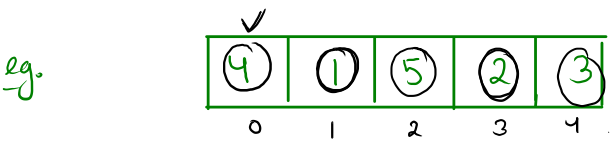
Sample Output 0

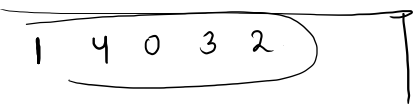
```
4 3 2 1 0
```





Try. -





Count Odd Pair

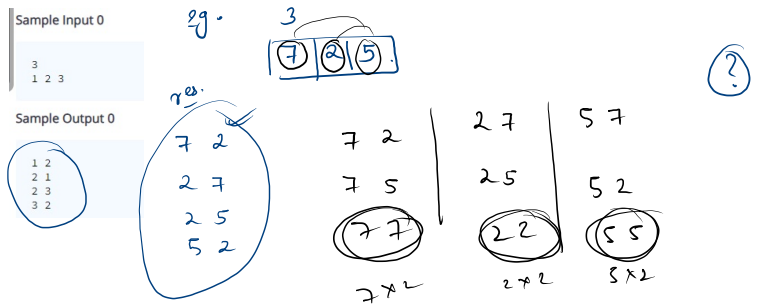
Take the array of size n and their values from user. And Find Pairs whose sum is odd.

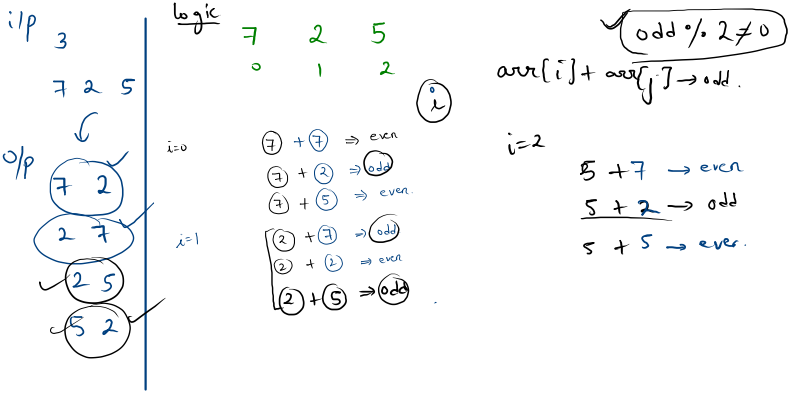
Sample Input 0

```
3
1 2 3
```

Sample Output 0

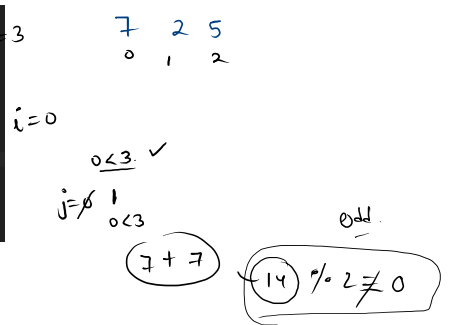
```
1 2
2 1
2 3
3 2
```






```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();  //size
    int [] arr = new int[n];
    for(int i = 0; i<n; i++){
        arr[i] = scn.nextInt();          // 7 2 5
    }

    for(int i = 0; i < n; i++){
        for(int j = 0; j<n; j++){
            if((arr[i] + arr[j]) % 2 != 0 ){
                System.out.println(arr[i] +" "+ arr[j]);
            }
        }
    }
}
```



Find Difference 1

Take the array and k as an integer input. Given condition is that the array contains all the unique elements. Then take the sum as an integer input and print all the permutations of the pairs whose absolute difference is k.

Sample Input 0

```
5
1 2 3 4 5
3
```

Sample Output 0

```
1 4
2 5
4 1
5 2
```

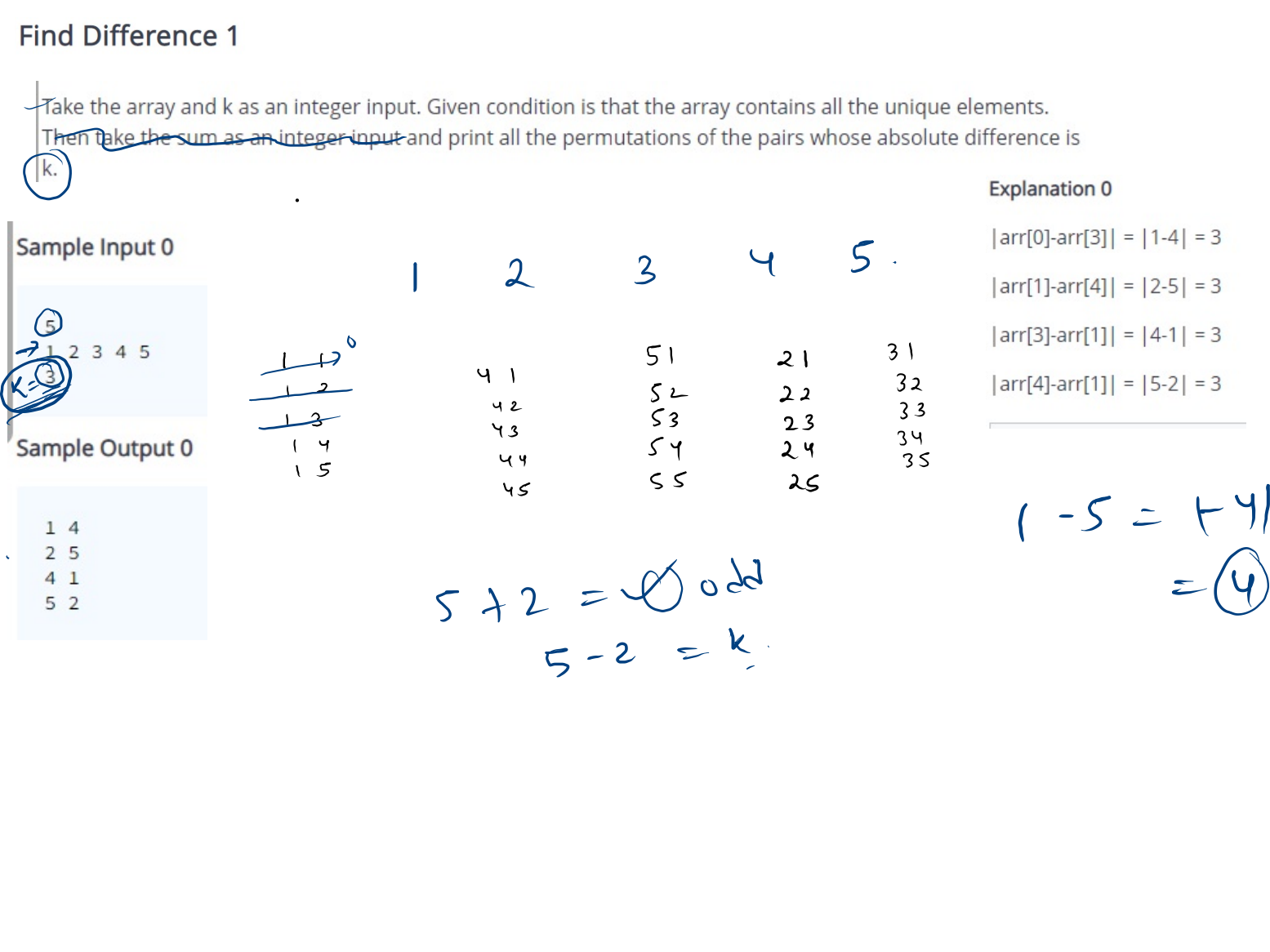
Explanation 0

|arr[0]-arr[3]| = |1-4| = 3

|arr[1]-arr[4]| = |2-5| = 3

|arr[3]-arr[1]| = |4-1| = 3

|arr[4]-arr[1]| = |5-2| = 3



Max Count 3

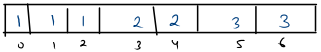
Take an array of size n with integer elements. And Print an element in the array which occurs for the maximum number of times.

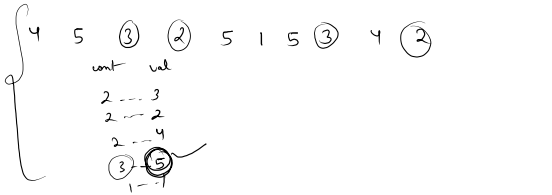
Sample Input 0

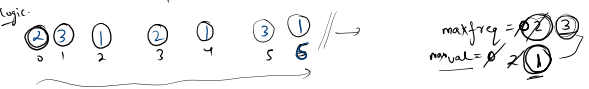
```
7
1 1 1 2 2 3 3
```

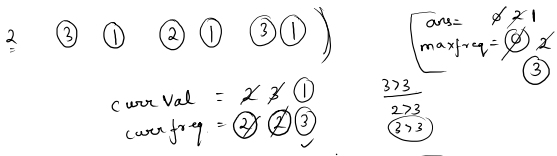
Sample Output 0

```
1
```







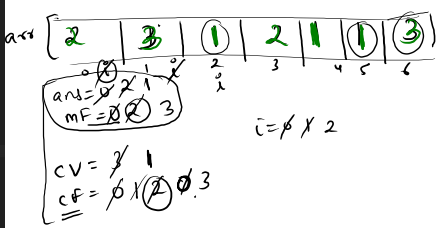



```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int [] arr = new int[n];
    for(int i = 0; i<n; i++){
        arr[i] = scn.nextInt();
    }

    int ans = 0;
    int maxFreq = 0;

    for(int i =0; i<n; i++){
        int currVal = arr[i];
        int currFreq = 0;
        for(int j = 0; j<n; j++){
            if(arr[j] == currVal){
                currFreq++;
            }
        }
        if(currFreq > maxFreq){
            ans = currVal;
            maxFreq = currFreq;
        }

    }
    System.out.println(ans);
```



return fun. ( )

{

}

p.s. int sum ()

{

return int.

}

int

0

!R, !P

void

```
public static void hello ( )
  {  Syso ("aman");
  }
```

R, !P

```
public static String hello ( )
  { return "thanks";
  }
```

!R, P

```
public static void (int age)
  {  Syso("Age is:" + age);
  }
```

R, P

```
ps int sum ( int a, int b)
  {   return a+b;
  }
```