Product of Elements Except Itself

2 5 3

0 1 2

ans → 15 6 10

↓

2 0 5 3

0 1 2 3

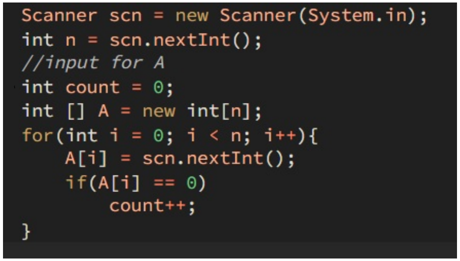
0 30 0 0

?

2 0 0 5 3

0 1 2 3 4

0 0 0 0 0



```
Scanner scn = new Scanner(System.in);
int n = scn.nextInt();
//input for A
int count = 0;
int [] A = new int[n];
for(int i = 0; i < n; i++){
    A[i] = scn.nextInt();
    if(A[i] == 0)
        count++;
}
```

2 0 5 3

count = 1

pwA = ~~1~~ 0

pwtO = ~~1~~ 30

2 0 5 3

c=1

20053

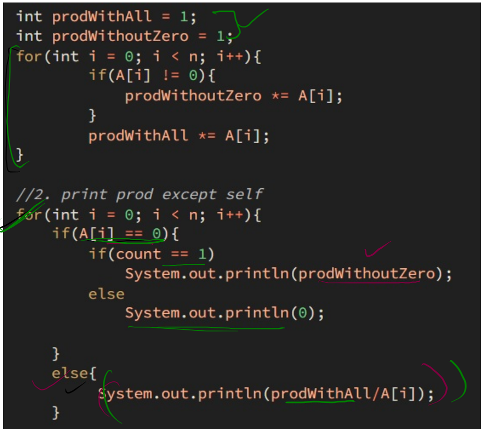
c=2

↓ ↓ ↓ ↓

2 0 0 5 3

0 1 2 3 4

0 0 0 0 0



```
int prodWithAll = 1;
int prodWithoutZero = 1;
for(int i = 0; i < n; i++){
    if(A[i] != 0){
        prodWithoutZero *= A[i];
    }
    prodWithAll *= A[i];
}

//2. print prod except self
for(int i = 0; i < n; i++){
    if(A[i] == 0){
        if(count == 1)
            System.out.println(prodWithoutZero);
        else
            System.out.println(0);

    }
    else{
        System.out.println(prodWithAll/A[i]);
    }
```

Solve Array.

~~4~~st ~~2~~ ~~3~~ 0

at ~~4th~~ index of ans array we have to put value ~~7~~ ~~8~~ ~~9~~ 10

11

val →	7	8	9	10	11
index →	4	1	2	~~3~~	~~0~~
	0	1	2	3	4
	i				

✓

ans.

11	8	9	10	7

$ans[index[i]] = val[i]$

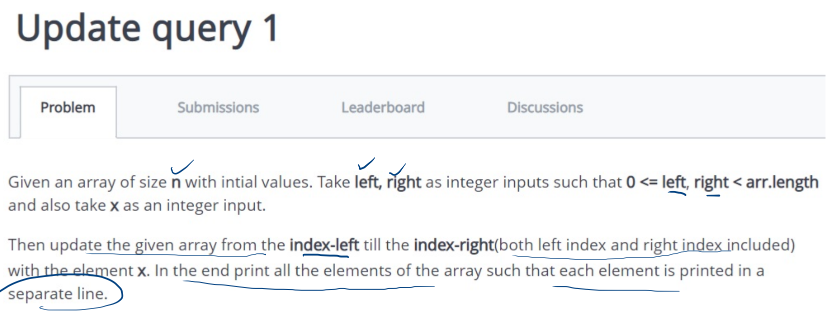
Check Characteristic.

$x < 0 \longrightarrow -1$

$x > 0 \longrightarrow 1$

$x = 0 \longrightarrow$ do nothing (0)

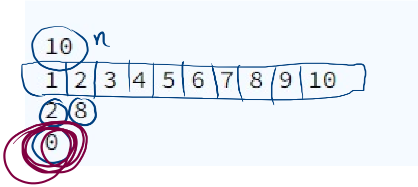
-1	0	1	1	-1	1
~~-7~~	~~0~~	~~1~~	~~4~~	~~-2~~	~~6~~



Update query 1

Given an array of size **n** with intial values. Take **left, right** as integer inputs such that **0 <= left, right < arr.length** and also take **x** as an integer input.

Then update the given array from the **index-left** till the **index-right**(both left index and right index included) with the element **x**. In the end print all the elements of the array such that each element is printed in a separate line.

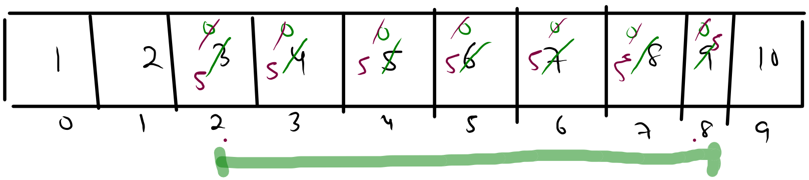


```
10
1 2 3 4 5 6 7 8 9 10
2 8
0
```

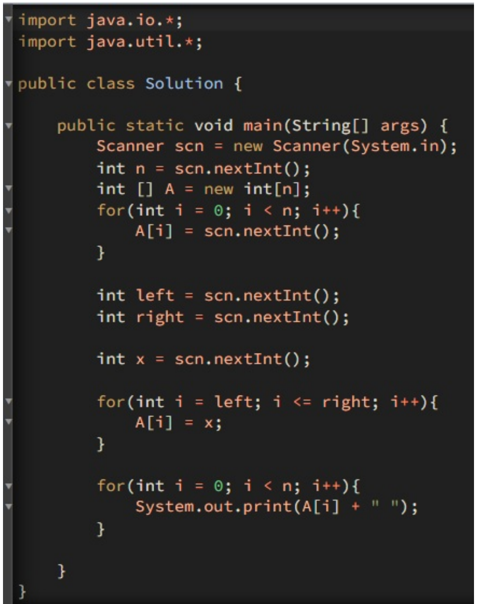
x = 0 / 5

left
right } valid index

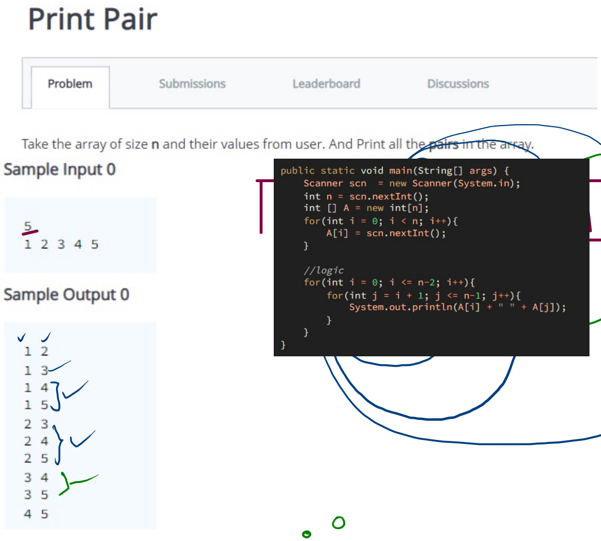
n = 10



l = 2
r = 8



```
import java.io.*;
import java.util.*;

public class Solution {

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

        int left = scn.nextInt();
        int right = scn.nextInt();

        int x = scn.nextInt();

        for(int i = left; i <= right; i++){
            A[i] = x;
        }

        for(int i = 0; i < n; i++){
            System.out.print(A[i] + " ");
        }

    }
}
```



Print Pair

Problem | Submissions | Leaderboard | Discussions

Take the array of size **n** and their values from user. And Print all the pairs in the array.

Sample Input 0

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

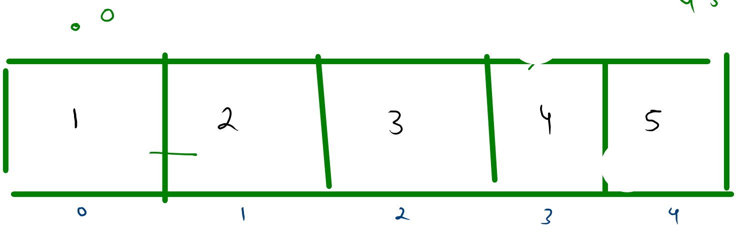
Sample Output 0

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

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

    //logic
    for(int i = 0; i <= n-2; i++){
        for(int j = i + 1; j <= n-1; j++){
            System.out.println(A[i] + " " + A[j]);
        }
    }
}
```

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

A[i] A[j] ✓
12 ✓
13 ✓
14 ✓
15 ✓
2 3 ✓
24 ✓
2 5 ✓
3 4 ✓
3 5 ✓
4 5



n=5

when j==n-1 update i → make j=i+1

i j

[0, n-2]

[i+1, n-1]

n=5

i	j
0,	1 2 3 4
1	2 3 4
2	3 4
3	4

$n=4$

1	2	3	4
0	1	2	3

$i=0$

$0 \le 2$ ✓

$i=1$

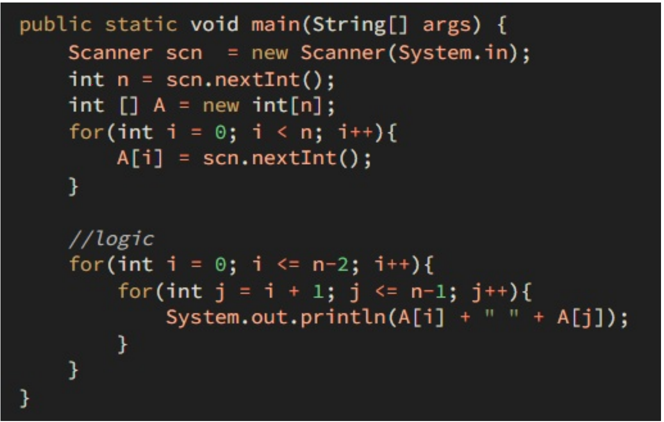
$i=2$

$2 \le 2$ ✓

$j=\cancel{3}.\ 4$

$3 \le 3$ ✓

$4 \le 3$ ✗



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

    //logic
    for(int i = 0; i <= n-2; i++){
        for(int j = i + 1; j <= n-1; j++){
            System.out.println(A[i] + " " + A[j]);
        }
    }
}
```

1 2

1 3

1 4

2 3

2 4

3 4

i+1 , n-1

[0,n-2]

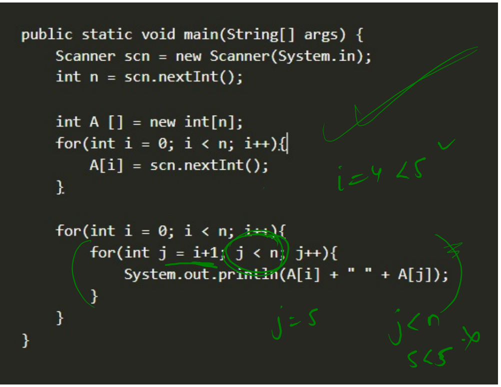
[0 , n-1)

n=5

[0, n-1)

[0, n-2]

[i+1, n-1]



```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();

    int A [] = new int[n];
    for(int i = 0; i < n; i++){
        A[i] = scn.nextInt();
    }

    for(int i = 0; i < n; i++){
        for(int j = i+1; j < n; j++){
            System.out.println(A[i] + " " + A[j]);
        }
    }
}
```

i=4 <5

j=5

j<n

5<5

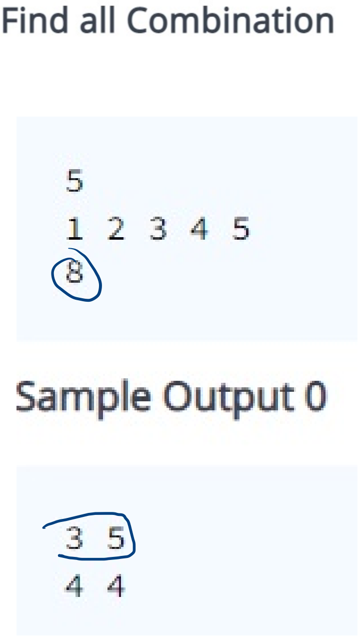
i= 0	1	2	3	4
1	2	3	4	
2	3	4		
3	4,			
4				

not reqr.

1	2	3	43	50
0	1	2	3	4

n=5

0	1	2	3	4
1	2	3	4	
2	3	4		
3	4			



Find all Combination

```
5
1 2 3 4 5
8
```

Sample Output 0

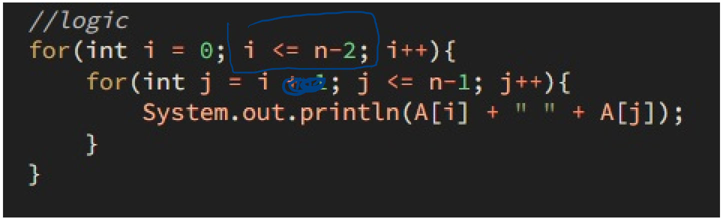
```
3 5
4 4
```

n=5

target=8

1	2	3	4	5
0	1	2	3	4

3 5
4 4



```
//logic
for(int i = 0; i <= n-2; i++){
    for(int j = i ~~+1~~; j <= n-1; j++){
        System.out.println(A[i] + " " + A[j]);
    }
}
```

3 3
3 4
3 5
5 5

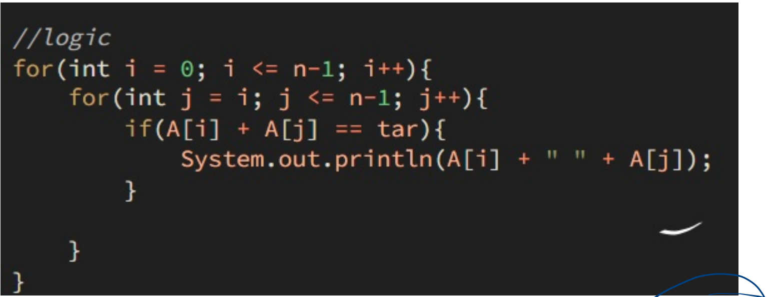
0 i

0 e

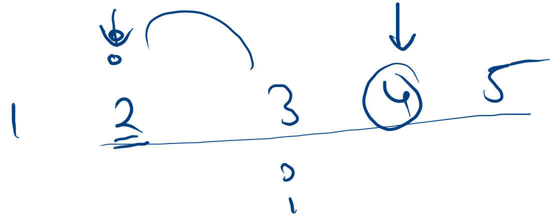
1 2 3 4 5

i j

3 3



```
//logic
for(int i = 0; i <= n-1; i++){
    for(int j = i; j <= n-1; j++){
        if(A[i] + A[j] == tar){
            System.out.println(A[i] + " " + A[j]);
        }

    }
}
```

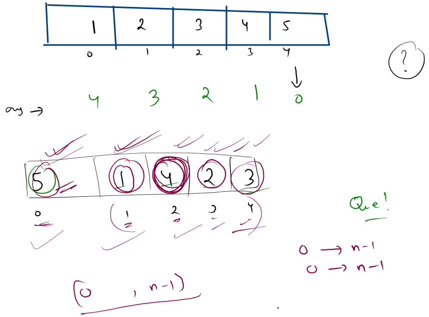


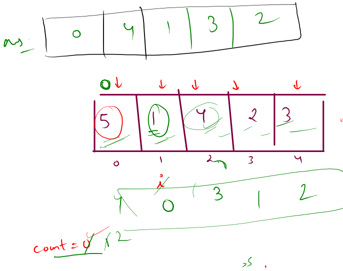
3 3
3 4
3 5

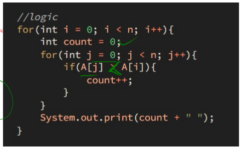
3 2
3 1

Greater Than Me

5
1 2 3 4 5



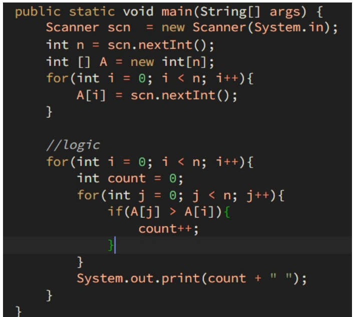




```
//logic
for(int i = 0; i < n; i++){
    int count = 0;
    for(int j = 0; j < n; j++){
        if(A[j] > A[i]){
            count++;
        }
    }
    System.out.print(count + " ");
}
```

1>1
4
j=0
2>5
3>5
A[0] > A[0]
1 >5
4>5
count = 0 4 5

5 1 4 3 2
4 1



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

    //logic
    for(int i = 0; i < n; i++){
        int count = 0;
        for(int j = 0; j < n; j++){
            if(A[j] > A[i]){
                count++;
            }
        }
        System.out.print(count + " ");
    }
}
```