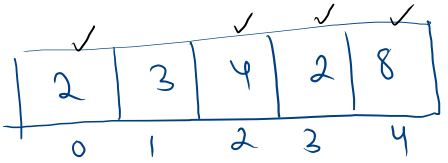
GKSTR35 Count_Even

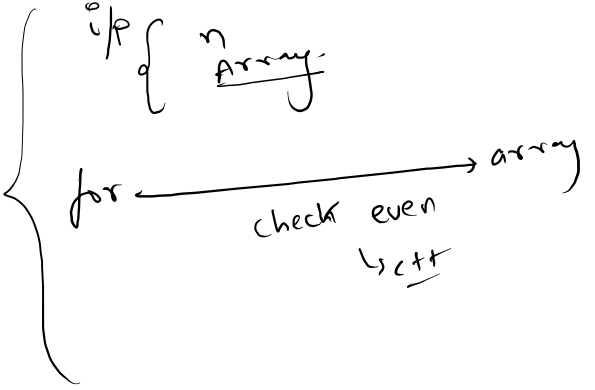
Given an integer **n**, the task is to define an integer array **arr[]** of size **n** &

Print the **Count / Number of even elements in the array.**

Sample Input 0

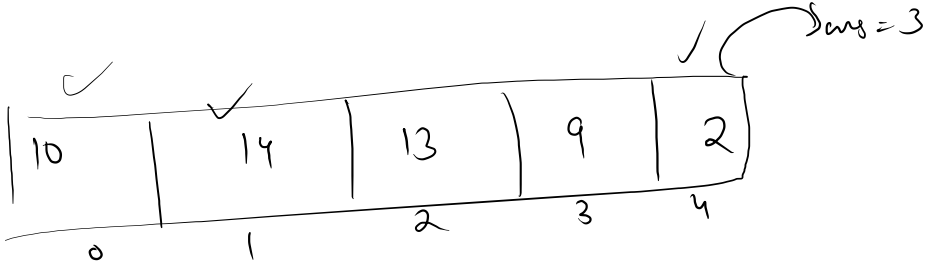
```
5
2
3
4
2
8
```

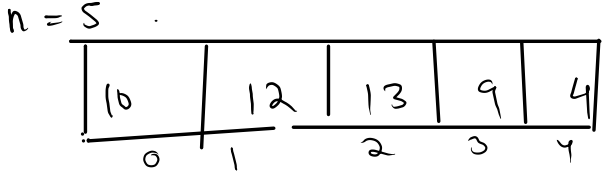




Sample Output 0

```
4
```



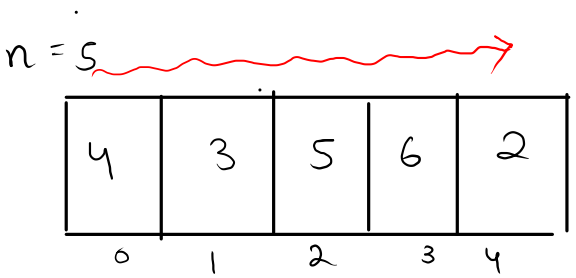


Ans = 3.

```java
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 count = 0;
        for(int i = 0; i < n; i++){
            if(A[i] % 2 == 0){
                count++;
            }
        }
        System.out.println(count);
    }
}
```

Maximum of Array

For the given array having **N** elements, find the **maximum element** of the array.



max = 6

logic. assume. max = A[0]

i

max = Math.max(max, A[i])

max = 4

i = 0 0 < 5 ✓

max = 4, 4

```java
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();
        }

        //max logic
        int max = A[0];
        for(int i = 0 ; i < n; i++){
            max = Math.max(A[i], max);
        }
        System.out.println(max);
    }
}
```

Check Characterstic

For each index,

Store 1 at that index if the element at that index is **greater than zero**.

Store 0 at the index if the element at that index is **equal to zero**.

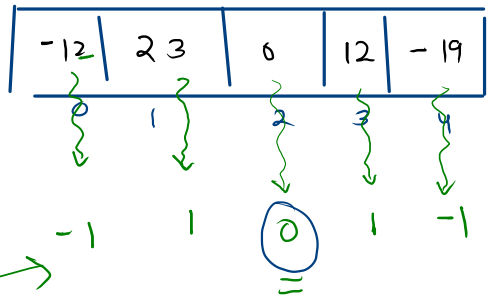
Store -1 at the index if the element at that index is **less than zero**.

Sample Input 0

```
5
-12 23 0 12 -19
```

Sample Output 0

```
-1 1 0 1 -1
```




```java
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();
        }

        for(int i = 0; i < n; i++){
            if(A[i] > 0){
                A[i] = 1;
            }else if(A[i] < 0){
                A[i] = -1;
            }
        }

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

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.

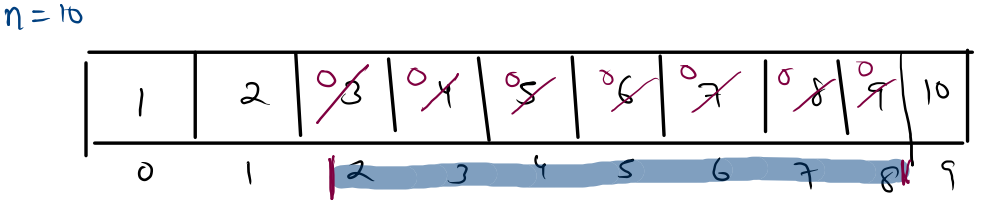
Then update the given array from the **index-left** till the **index-right**(both left index and right index included) with the element **x**. In the end print all the elements of the array such that each element is printed in a separate line.

Sample Input 0

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

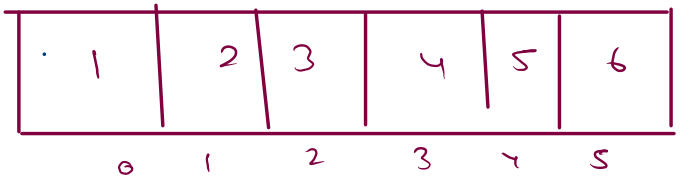
Sample Output 0

```
1 2 0 0 0 0 0 0 0 10
```



```java
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 l = scn.nextInt();
        int r = scn.nextInt();
        int x = scn.nextInt();

        //logic
        for(int i = l; i <= r; i++){
            A[i] = x;
        }
        //print
        for(int i = 0; i < n; i++){
            System.out.print(A[i] + " ");
        }
    }
}
```

$n = 6$



$l = 1$

$r = 3$

$x = 5$

Product of Elements Except Itself

Declare the first array of size **n** that stores values of int data-type. Then take **n** integer inputs and store them in the array one by one.

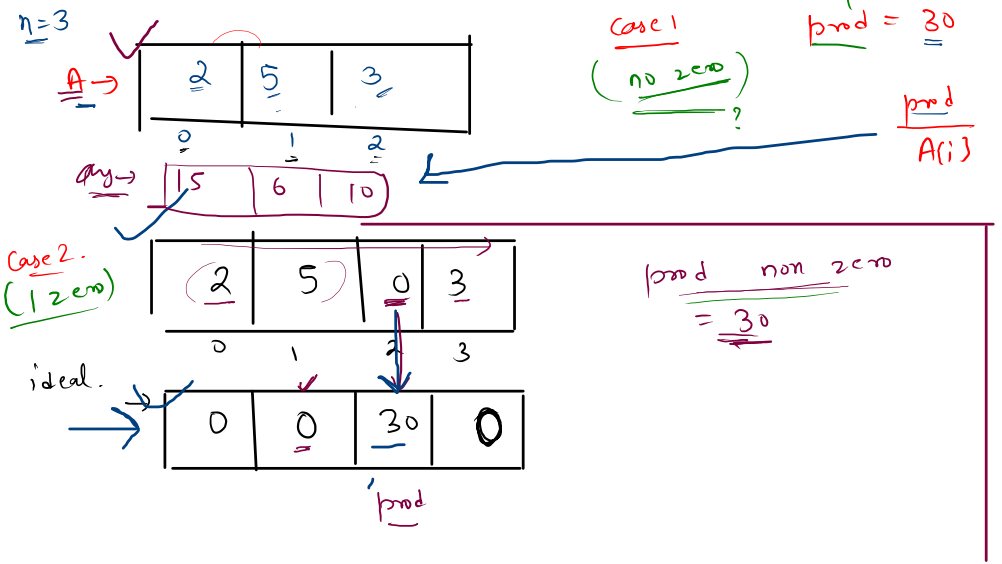
For each index print the **product** of all the elements except the element present at that index..

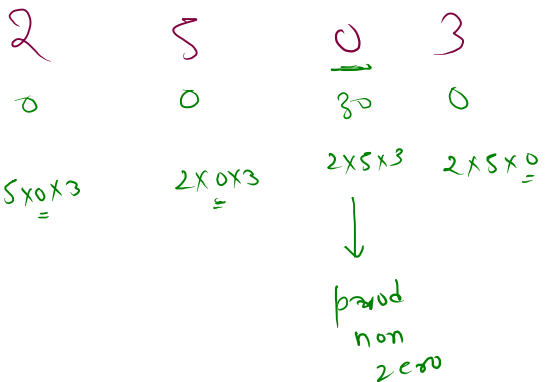
Sample Input 0

```
3
2
5
3
```

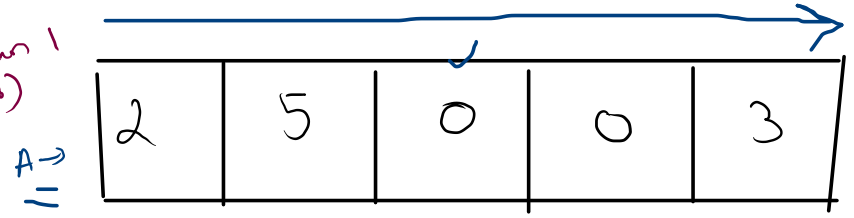
Sample Output 0

```
15
6
10
```





Case 3
(more than 1 zero)



5 x 0	2	2	2	2
x 0	x	x 5	5	5
x 3	0	x 0	0	6
	x 0	x 3	3	0
	x 3			

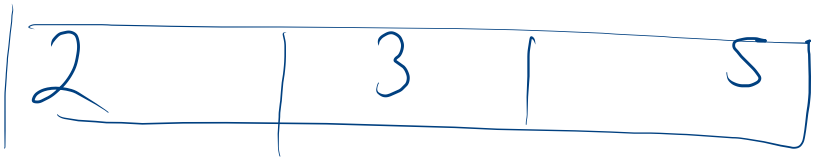
0 0 0 0 0

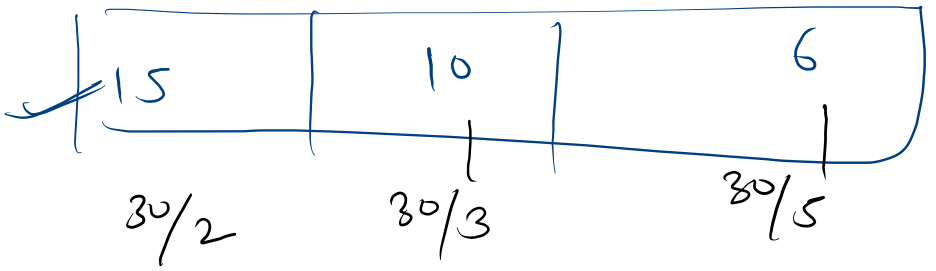
***

1. → How many zero
2. → prod of non zero

zero = 0

odd = 30





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

public class Solution {
    public static int [] productExceptSelf(int [] A){
        int [] ans = new int[A.length];

        //calc 2 imp factor
        int zero = 0;
        int prod = 1;   //prod of non zero
        for(int i = 0; i < A.length; i++){
            if(A[i] == 0){
                zero++;
            }else{      // non zero
                prod *= A[i];
            }
        }

        //case 1: zero = 0
        if(zero == 0){
            for(int i = 0; i < ans.length; i++){
                ans[i] = prod/A[i];
            }
        }
        //case 2: zero = 1
        else if(zero == 1){
            for(int i = 0; i < ans.length; i++){
                if(A[i] == 0){
                    ans[i] = prod;
                }
            }
        }
        return ans;

    }
    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 [] ans = productExceptSelf(A);

        for(int i = 0; i < ans.length; i++){
            System.out.println(ans[i]);
        }

    }
}
```