Note:-

when trying to find max value, initially you should have $-\infty$

```
Integer.MIN_VALUE;
```

when trying to find min value, initially you should have $+\infty$

```
Integer.MAX_VALUE;
```

Product of Elements Except Itself

nested loops
when you travel entire array for each element

$n = 6$

	0	1	2	3	4	5
arr =	2	3	1	2	5	1

	0	1	2	3	4	5
ans =	30	20	60	30	12	60

means, for each 'i' index, traverse from 0 to n-1 and skip 'i' index and find product.

i = 0, j = 0 → 5 and skip 0
i = 1, j = 0 → 5 and skip 1
i = 2, j = 0 → 5 and skip 2
i = 3, j = 0 → 5 and skip 3
i = 4, j = 0 → 5 and skip 4
i = 5, j = 0 → 5 and skip 5

Brute force / Naive :- most basic approach

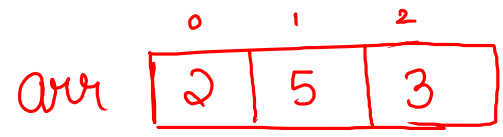
code

```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();
    }

    prodExceptItself(arr, n);
}
public static void prodExceptItself(int[] arr, int n){
    for (int i = 0; i < n; i++) {
        int prod = 1;
        for (int j = 0; j < n; j++) {
            if ( i != j ) {
                prod = prod * arr[j];
            }
        }
        System.out.println(prod);
    }
}
```

o/p

15
6
10



i = 0, prod = 1
j = 1, prod = 5
j = 2, prod = 15

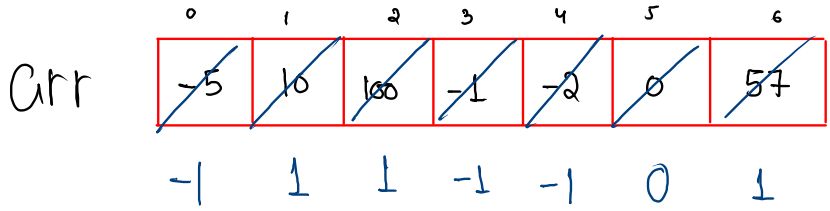
i = 1, prod = 1
j = 0, prod = 2
j = 2, prod = 6

i = 2, prod = 1
j = 0, prod = 2
j = 1, prod = 10

⟹ Updating array elements

```
arr[i] = val;
```

Check Characterstic




```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 = checkChar(arr, n);
    for (int i = 0; i < n; i++) {
        System.out.print(ans[i] + " ");
    }
}
public static int[] checkChar(int[] arr, int n) {
    for (int i = 0; i < n; i++) {
        if ( arr[i] > 0 ) {
            arr[i] = 1;
        } else if ( arr[i] == 0 ) {
            arr[i] = 0;
        } else if ( arr[i] < 0 ) {
            arr[i] = -1;
        }
    }

    return arr;
}
```

Solve Array

$n = 5$

	0	1	2	3	4
arr1	10	11	12	13	14

	0	1	2	3	4
arr2	2	1	4	0	3

Note:- we are trying to print values of 1st array, on the basis of value of 2nd array, as an index

Note:- $val = arr1[arr2[i]]$

o/p)

$i = 0$, $val = 12$

$i = 1$, $val = 11$

$i = 2$, $val = 14$

$i = 3$, $val = 10$

$i = 4$, $val = 13$

wrong code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr1 = new int[n];
    for (int i = 0; i < n; i++) {
        arr1[i] = scn.nextInt();
    }

    int[] arr2 = new int[n];
    for (int i = 0; i < n; i++) {
        arr2[i] = scn.nextInt();
    }

    solve(arr1, arr2, n);
}
public static void solve(int[] arr1, int[] arr2, int n) {
    int[] ans = new int[n];
    for (int i = 0; i < n; i++) {
        ans[i] = arr1[arr2[i]];
    }

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