

Check Characteristic

Example

5				
-12	23	0	12	-19
0	1	2	3	4

-1	+1	0	+1	-1
----	----	---	----	----

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

Solve Array

Take n as an integer input representing size of both array.

Take n integer inputs for numbers array and Then take n integer inputs for array indexes where each integer input can be from 0 till numbers.length.

Then create an array of size n and name it target array. From left to right read numbers[i] and index[i], and in the target array at the index index[i], insert the value numbers[i].

int n = 5;

numbers =

10	12	15	17	8
0	1	2	3	4

indexes =

3	0	2	1	4
0	1	2	3	4

values
0 → 4
0 → 4

indexes

target =

12	17	15	10	8
0	1	2	3	4

```

public static void main(String[] args) {
    /* Enter your code here. Read input from STDIN. Print out
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();

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

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

    solveArray(n, num, idx);
}

```

operation = $n + n$
 $T.C = O(n)$
 $S.C = O(n)$

```

public static void solveArray(int n, int[] num, int[] idx) {

```

```

    → int[] target = new int[n];
    → for (int i = 0; i < n; i++) {
        int value = num[i];
        int index = idx[i];
        target[index] = value;
    }
    → for (int i = 0; i < n; i++) {
        System.out.print(target[i] + " ");
    }
}

```

$target[idx[i]] = num[i];$

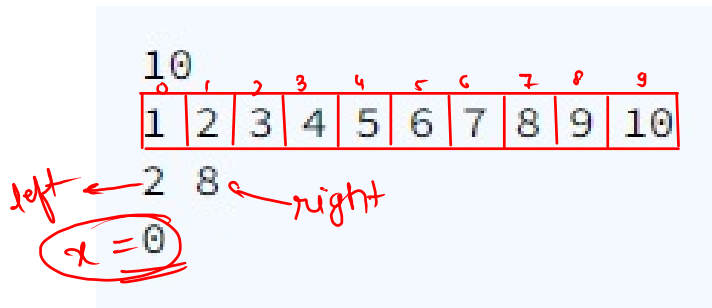
Update query 1

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

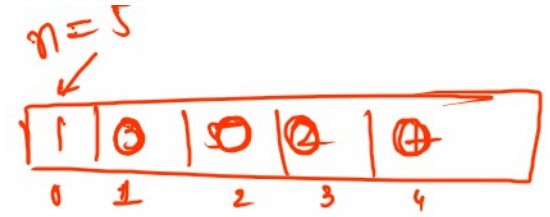
    update(arr, n, left, right, x);
}

public static void update(int[] arr, int n, int left, int right, int x) {
    for (int i = left; i <= right; i++) {
        arr[i] = x;
    }

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



i = 2, arr[2] = 0
i = 3, "
i = 4, "
i = 5, "
i = 6, "
i = 7, "
i = 8, "



Add One

arr:

1	2	3	8
0	1	2	3

 } +1 return // stop

only 2 conditions possible:-

→ 0 to 8 ⇒ +1 and return
 → 9 ⇒ place 0 in that index and move back with carry = 1

1 2 3 9
 + 1
~~1 2 3 9~~
 0

dry run

arr:

2	9	9	9	9	0
0	1	2	3	4	5

i = 5, arr[5] = 0

i = 4, arr[4] = 0

i = 3, arr[3] = 0

i = 2, arr[2] = 0

i = 1, arr[1] = 0

i = 0, arr[0] = 2

0 → 8

return;

n = 1

0
0

arr ⇒

1

i = 0, arr[0] = 1
 return;

n = 3

9	9	9
0	1	2

i = 2, arr[2] = 0;

i = 1, arr[1] = 0

i = 0, arr[0] = 0

1	0	0	0
---	---	---	---

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

```
public static int[] plusOne(int[] arr, int n) {
```

```
1) for (int i = n - 1; i >= 0; i--) {
    2) if (arr[i] < 9) { // 0-8
        arr[i]++;
        return arr;
    }
    arr[i] = 0;
```

```
int[] ans = new int[n + 1];
ans[0] = 1;
return ans;
}
```

e.g.,

1	8	9	9
0	1	2	3

i = 3,

1	8	9	0
---	---	---	---

i = 2,

1	8	0	0
---	---	---	---

i = 1,

1	9	0	0
---	---	---	---

main

e.g.,

9	0	9
0	1	2

→ 910

i = 2,

9	0	0
0	1	2

i = 1,

9	1	0
---	---	---