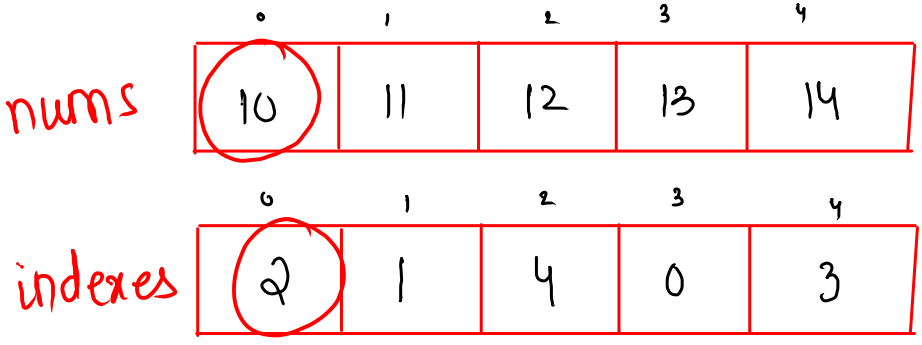
Solve Array

n = 5



int idx = indexes[i];
int val = nums[i];

target[idx] = val;

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]**.

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

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

    solve(nums, indexes, n);
}
public static void solve(int[] nums, int[] indexes, int n) {
    int[] target = new int[n];

    // main logic
    for (int i = 0; i < n; i++) {
        int val = nums[i];
        int idx = indexes[i];

        target[idx] = val;
    }

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

Page 22

2nd question

Update query 1

$n = 8$

arr

1	5	2	-4	0	7	10	15
0	1	2	3	4	5	6	7

left = 4

right = 7

x = 1

traverse from left index to right index and update all current values with 'x' variable

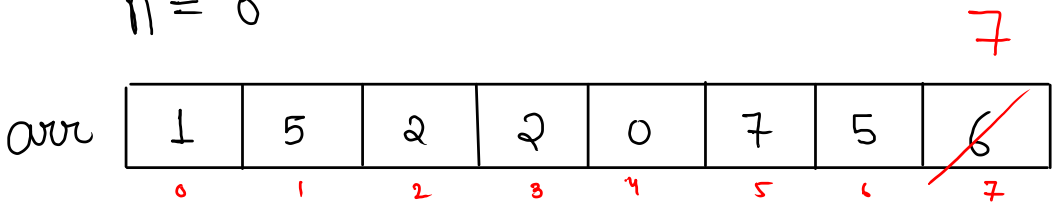
```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n; i++) {
        arr[i] = scn.nextInt();
    }
    int left = scn.nextInt();
    int right = scn.nextInt();
    int x = scn.nextInt();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

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

Add One

(gmp) M2

n = 8



15220756
+1
15220757

arr1

2	3	9	9

2399
+1
2400

1) 12345 → 12346

2) 32545 → 32545

3) 12349 → 12350

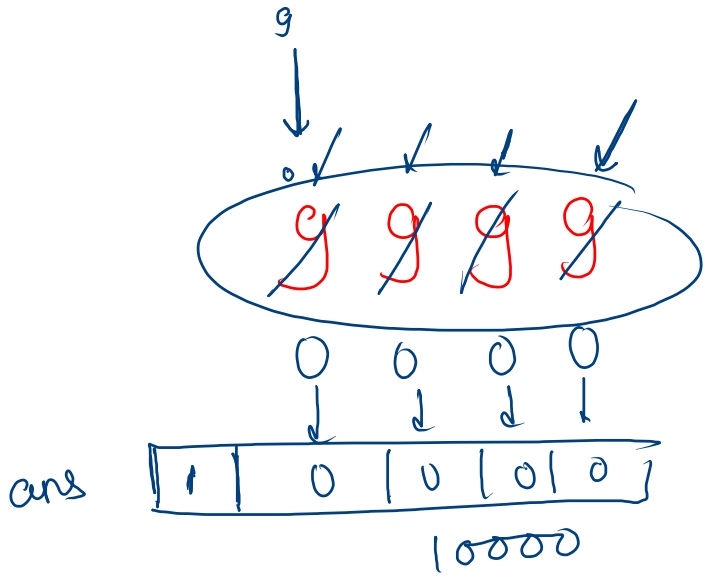
4) 1299 → 1300

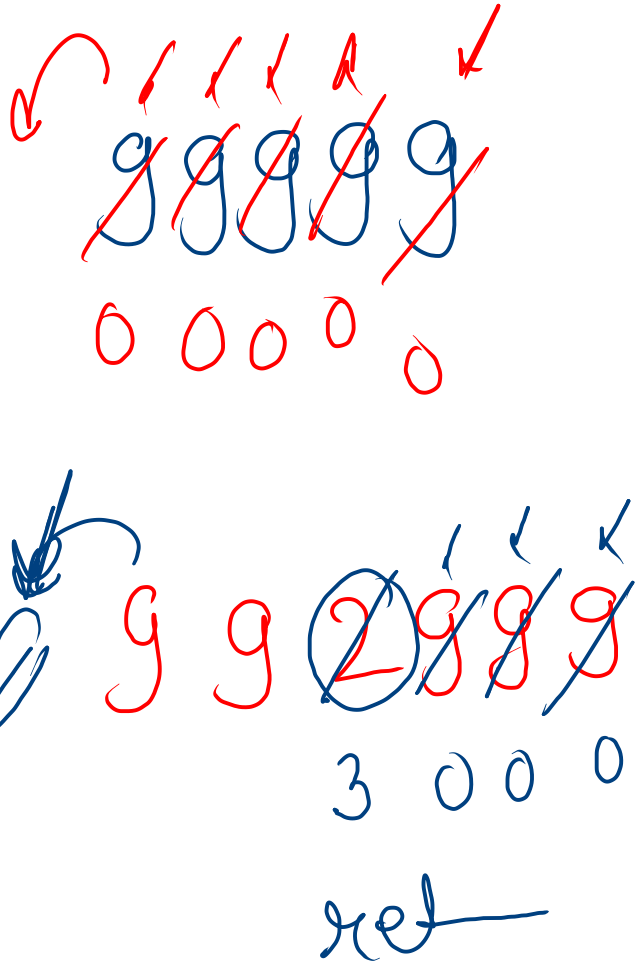
if num is 9, then update with '0'
else, increase by 1

5) 9999 → 10000

Note :-

traverse n-1 to 0,
if current element is 9 then update it with zero
else, increment current element by 1 and end the process.





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();
    }

    int[] ans = addOne(arr, n);
    for (int i = 0; i < ans.length; i++) {
        System.out.print(ans[i] + " ");
    }
}
// main logic
public static int[] addOne(int[] arr, int n) {

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

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