Update query 1

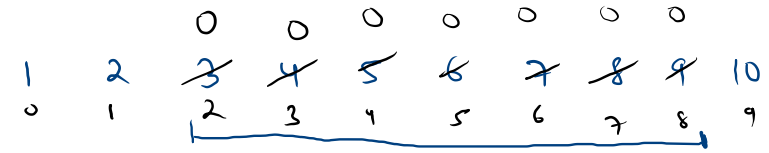
Given an array of size **n** with intial values. Take **left, right** as integer inputs such that **0 <= left, right < arr.length** and also take **x** as an integer input.

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.

0 ≤ left, right < n

left
right
x

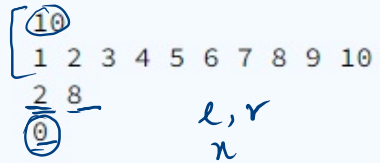
n = 10



l = 2
r = 8
x = 0

1. i/p
2. move [l, r] → update.
3. print

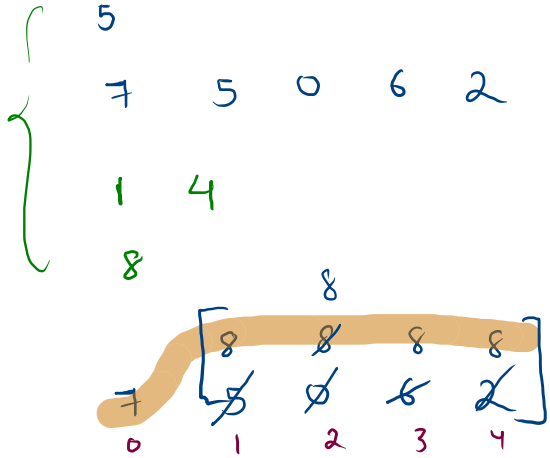
Sample Input 0



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 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] + " ");
        }

    }
}
```



l = 1
r = 4
x = 8

i = 1 1 ≤ 4
= 2 2 ≤ 4 ✓
= 3 3 ≤ 4 ✓
= 4 4 ≤ 4 ✓
5 ≤ 4 ✗

A[2] = 8

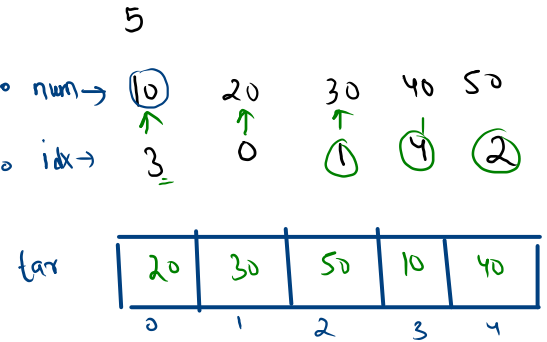
Solve Array

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**.

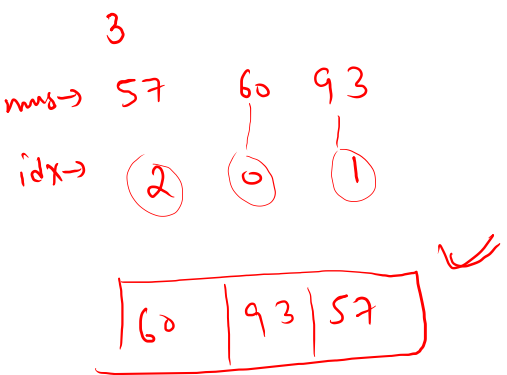
Then create an array of size n and name it target array. From left to right read **numbers[i] and index[i]**, and in the target array at the index **index[i]**, insert the value **numbers[i]**.

: target[index[0]] = numbers[0], i.e target[0]= 12.

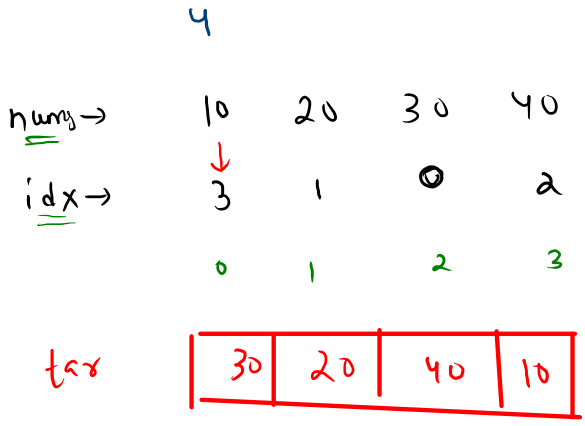


tar [3] = 10

tar [idx[i]] = num[i].



```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 [] nums = new int[n];
        for(int i = 0 ;i < n; i++){
            nums[i] = scn.nextInt();
        }
        int [] idx = new int[n];
        for(int i = 0 ;i < n; i++){
            idx[i] = scn.nextInt();
        }
        //logic
        int [] tar = new int[n];
        for(int i = 0 ;i < n; i++){
            tar[idx[i]] = nums[i];
        }

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

    }
}
```



i=0 0<4 ✓ tar[3] = 10

i=1 1<4 tar[1] = 20

i=2 2<4 ✓ tar[0] = 30

i=3 3<4 tar[2] = 40

i=4 (4<4) ✗

Add One

Problem | Submissions | Leaderboard | Discussions

Take an **array arr** of size **N** as input which represents a **large number**.

Add 1 (one) to this large number and print the resultant array.

eg:- [4,2,3,6,5,8,7,1,5,3,9,6] In this case answer must be [4,2,3,6,5,8,7,1,5,3,9,7]

Note: The large integer does not contain any **leading 0's** in the array.

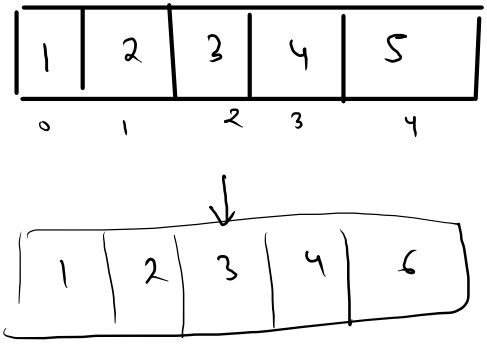
NOTE:- After answering the question, attempt the related question in the linked resource to improve your understanding of this question . Click here

Sample Input 0

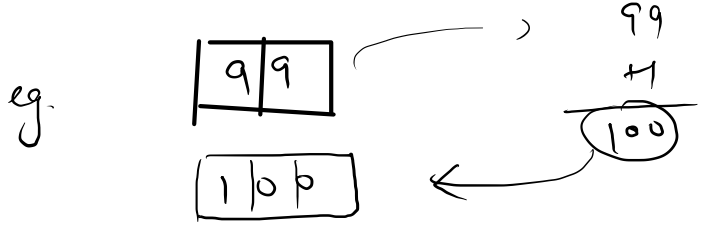
```
5
1 2 3 4 5
```

Sample Output 0

```
1 2 3 4 6
```

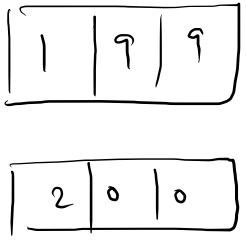


12345
+ 1
12346

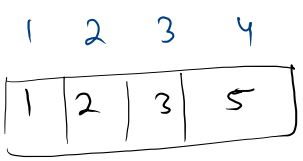


eg.3.

199
+1
200

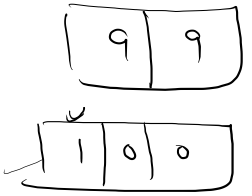


eg 1.



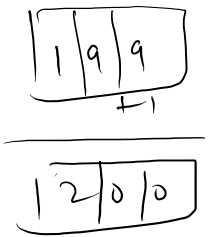
1234 +1

eg 2.



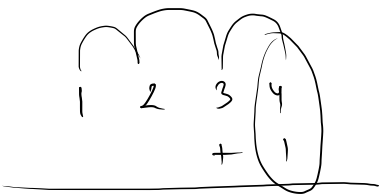
99 +1 100

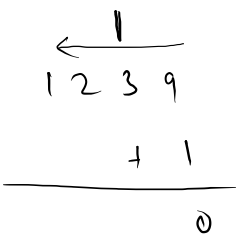
eg



56

```
0 <= arr[i] <= 9
```



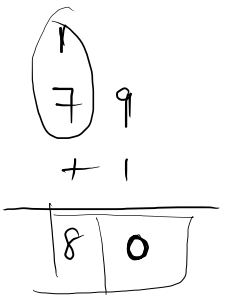


1. $0 \le A[i] \le 9$ +1

at max A[i]+1 ?

$= 10$

eg.



1. $0 \leq A[i] \leq 9$

at max sum

$A[i] + 1 = 10$

$$\begin{array}{r} {\scriptstyle 0} \\ 7\ 6 \\ +\ 1 \\ \hline 7\ 7 \end{array}$$

$6 + 1 = 7$

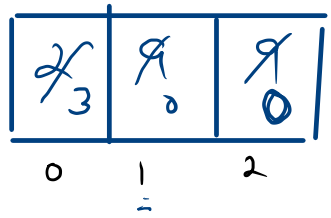
$$\begin{array}{r} {\scriptstyle 1} \\ 7\ 9 \\ +\ 1 \\ \hline 0 \end{array}$$

2. $\begin{array}{r} 7\ 6 \\ \hline \end{array}$ carry = 1

$\begin{array}{r} 7\ 9 \\ \hline \end{array}$ carry = 1

```
14          int carry = 1;
15          for(int i = n-1; i>= 0; i--){
16              int val = A[i] + carry;
17              if(val == 10){
18                  A[i] = 0;
19                  carry = 1;
20              }else{  //single digit
21                  A[i] = val;
22                  carry = 0;
23              }
24          }
25
26      }
```

n=3



C = 1 1 1 0

i=2 220

val = 9+1 = 10

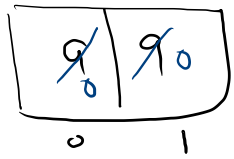
i=1 v = 9+1 = 10

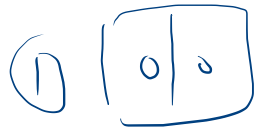
i=0 v = 2+1 = 3

i=-1

```
        int carry = 1;
        for(int i = n-1; i>= 0; i--){
            int val = A[i] + carry;
            if(val == 10){
                A[i] = 0;
                carry = 1;
            }else{
                A[i] = val;
                carry = 0;
            }
        }

    }
```

C=1





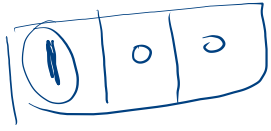
i=1 $1 \geq 0$ ✓

val = 9+1 = 10

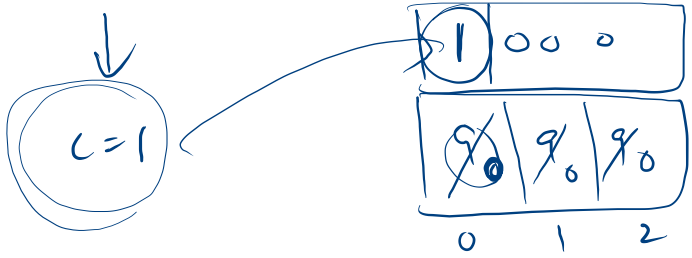
i=0 $0 \geq 0$ ✓

val = 10

i=-1 $-1 \geq 0$ ✗



```
        int carry = 1;
        for(int i = n-1; i>= 0; i--){
            int val = A[i] + carry;
            if(val == 10){
                A[i] = 0;
                carry = 1;
            }else{
                A[i] = val;
                carry = 0;
            }
        }

    }
```



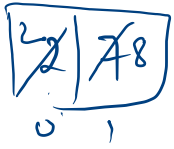
i=2 220 val=10

i=1 120 val=10

i=0 020 val=10

i=-1 -1>=0 ✗

carry=1 0



i=1

i=0 val=2+0

i=-1

val=7+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();
        }

        int carry = 1;
        for(int i = n-1; i>= 0; i--){
            int val = A[i] + carry;
            if(val == 10){
                A[i] = 0;
                carry = 1;
            }else{
                A[i] = val;
                carry = 0;
            }
        }

        if(carry == 1){
            A = new int[n+1];
            A[0] = 1;
        }
        for(int i = 0 ; i < A.length; i++){
            System.out.print(A[i] + " ");
        }

    }
}
```

Print Pair

nested loops with Array.

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
```

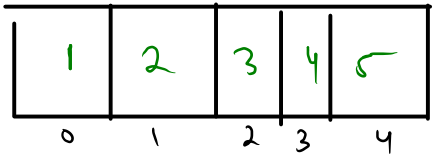
i j

0 { 1 2 → 1, 1 3 → 2, 1 4 → 3, 1 5 → 4

1 { 2 3 → 2, 2 4 → 3, 2 5 → 4

2 { 3 4 → 3, 3 5 → 4

3 { 4 5 → 4



n = 5

index.

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

i [0, n-2]

j = [i+1 — n-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();
        }
        //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]);
            }
        }

    }
}
```

eg. n=4

10 20 30 40

10 20

10 30

10 40

20 30

20 40

30 40