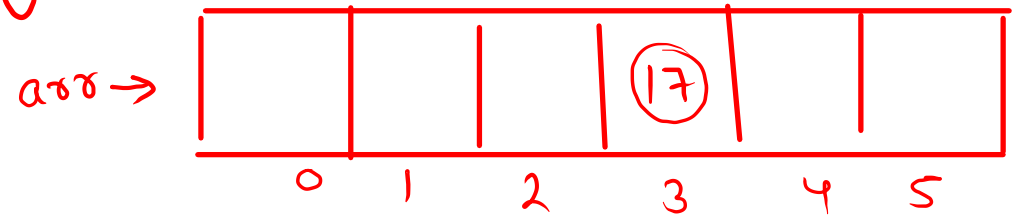
Array

size = 6




Sys o (arr[3]);

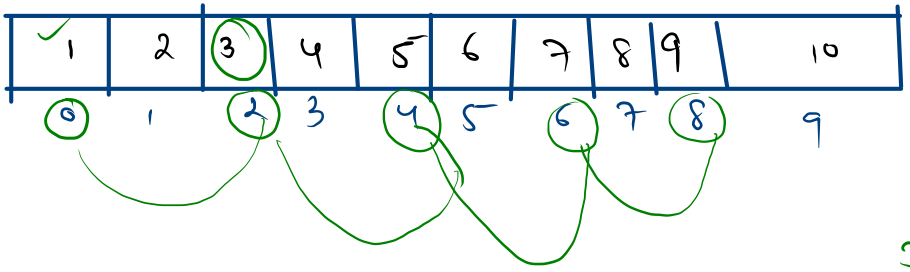
# Print Alternate Array Elements Linewise

Problem | Submissions | Leaderboard | Discussions

Take n as an integer input. Declare an **array** of size **n** that stores value of **int** data-type. Then take **n** integer inputs and store them in the array one by one.

Then print the alternate elements of the array starting from the **0th** index

n=10




idx even → val

1
3
5
7
9

for (i=0 ... n-1 ; i+=2)
{
}

**Sample Input 0**

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

**Sample Output 0**

```
1
3
5
7
9
```


```
6   public static void main(String[] args) {
7       Scanner scn = new Scanner(System.in);
8       int n = scn.nextInt();
9       int [] A = new int[n];
10
11      for(int i = 0; i < n; i++){
12          A[i] = scn.nextInt();
13      }
14
15      for(int i = 0; i < n; i += 2){
16          System.out.println(A[i]);
17      }
18
```

0 < 10 ✓

2 < 10 ✓

i = 0 2

1

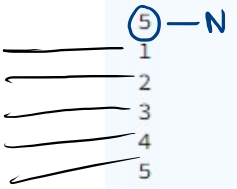
# Print Array Elements Reverse linewise

Problem | Submissions | Leaderboard | Discussions

Take N as an integer input. Declare an **array** of size **N** that stores value of int data-type. Then take N integer inputs and store them in the array one by one.

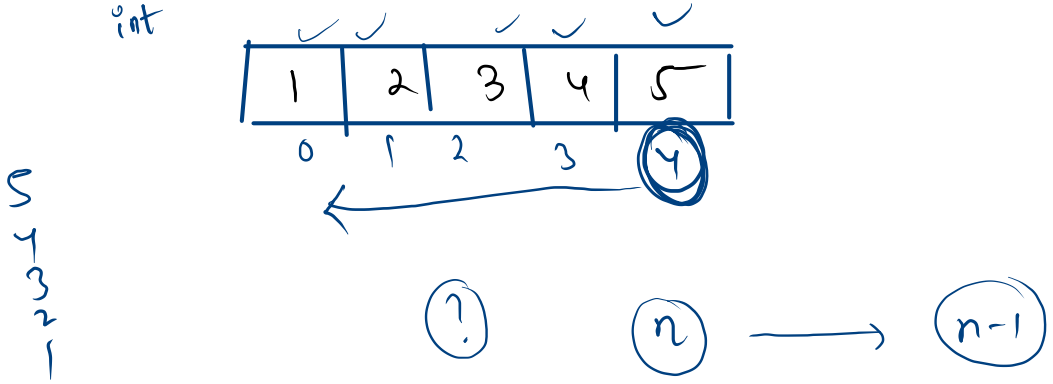
Then print the elements of the array from the last index till the **0th** index such that each element is printed one by one in each line

## Sample Input 0

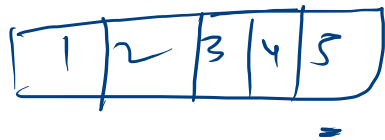



## Sample Output 0

```
5 4 3 2 1
```



```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int N = scn.nextInt();

    //declare arr
    int [] A = new int[N];

    //input
    for(int i = 0; i < N; i++){
        A[i] = scn.nextInt();
    }

    //output
    for(int i = N-1; i >= 0; i--){
        System.out.print(A[i] + " ");
    }

}
```

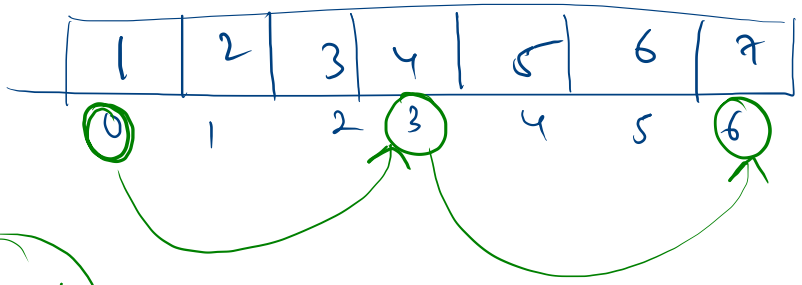


# Print Array element if index divisible by 3

Problem | Submissions | Leaderboard | Discussions

Take **n** as an integer input. Declare an **array** of size **n** that stores value of **int** data-type. Then take **n** integer inputs and store them in the array one by one.

Then print all the elements of the array where the **index** is **divisible by 3**




i = 0 ✓
+3 ✓

```
if (i%3==0)
{
    System(A[i])
}
```

## Sample Input 0

```
7
1
2
3
4
5
6
7
```

n

## Sample Output 0

```
1 4 7
```

Divisible by 3.

```
3
4  public class Solution {
5
6      public static void main(String[] args) {
7          Scanner scn = new Scanner(System.in);
8          int n = scn.nextInt();
9          int [] A = new int[n];
10
11         for(int i = 0; i < n; i++){
12             A[i] = scn.nextInt();
13         }
14
15         for(int i = 0; i < n; i += 3){
16             System.out.print(A[i] + " ");
17         }
18
19     }
20 }
```

if

i % 3 == 0

# Check if two arrays are identical?

Problem | Submissions | Leaderboard | Discussions

Take **n** as an integer input. 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.

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

Then print **true** if the arrays are equal and print **false** if the array is not equal.

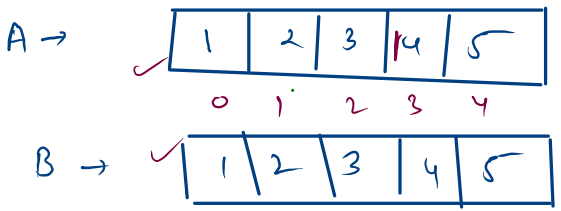
**Definition of Equal Arrays:** Arrays whose size is equal and whose elements at the corresponding indexes are the same

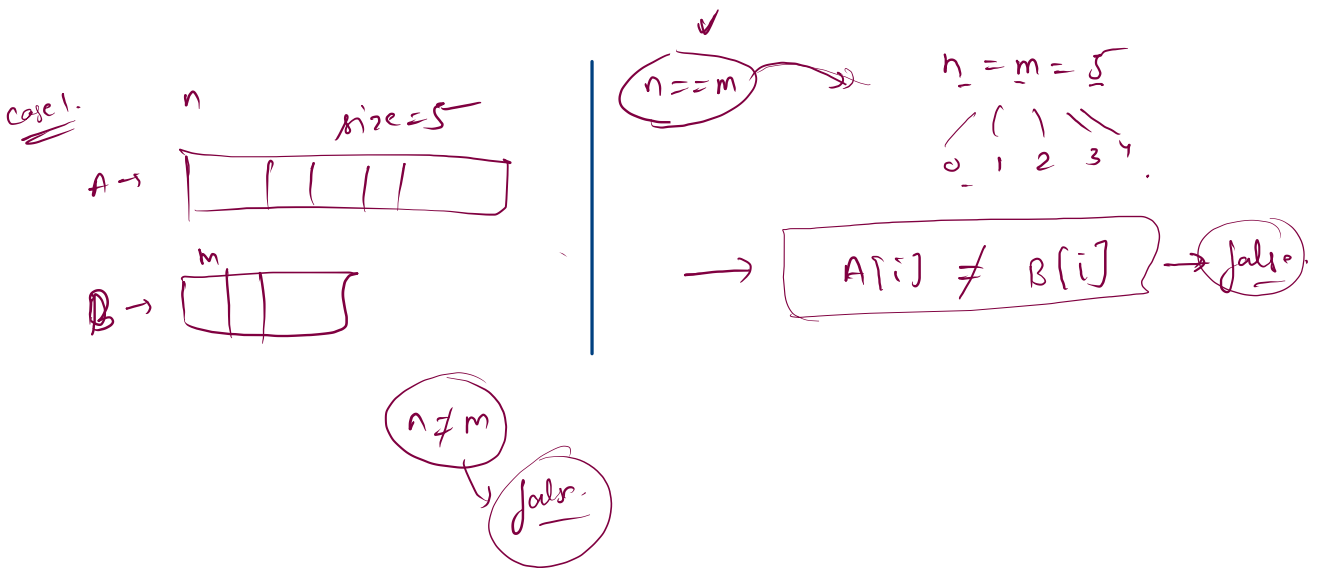
## Sample Input 0

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

## Sample Output 0

```
true
```



ans = true / false

n = 3

assuming.

boolean even = true

$n \neq m$

true.

if ( n % 2 == 0 )

{

even = false

}

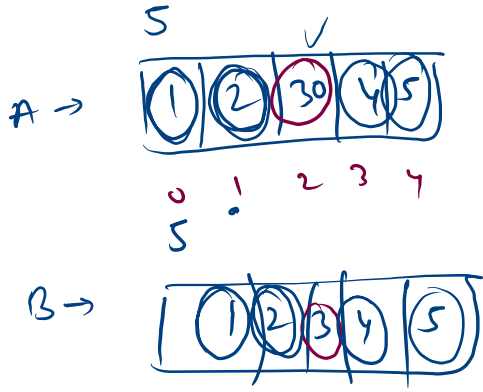
ans = false / true

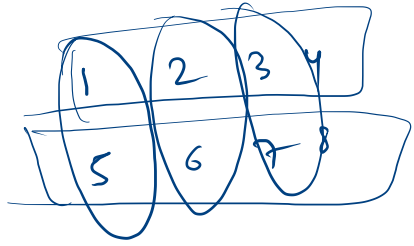
true.

true.

```java
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 m = scn.nextInt();
        int [] B = new int[m];
        for(int i = 0 ; i < m; i++){
            B[i] = scn.nextInt();
        }

        //
        boolean ans = true; // true means identical

        if(n != m){
            ans = false;
        }
        else{
            //compare : n == m
            for(int i = 0 ; i < n; i++){
                if(A[i] != B[i]){
                    ans = false;
                }
            }

        }
        System.out.println(ans);


```

```
20          if(n != m){
21              System.out.println("false");
22          }
23          else{
24              //compare : n == m
25              for(int i = 0 ; i < n; i++){
26                  if(A[i] != B[i]){
27                      System.out.println("false");
28                      return;
29                  }
30              }
31              System.out.println("true");
32          }
33
34      }
35  }
```



break → come out of loop

return → " " " function.

# Print two arrays alternately

Problem | Submissions | Leaderboard | Discussions

Take **n** as an integer input. 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.

Declare the **second 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.

Then print the elements as explained below

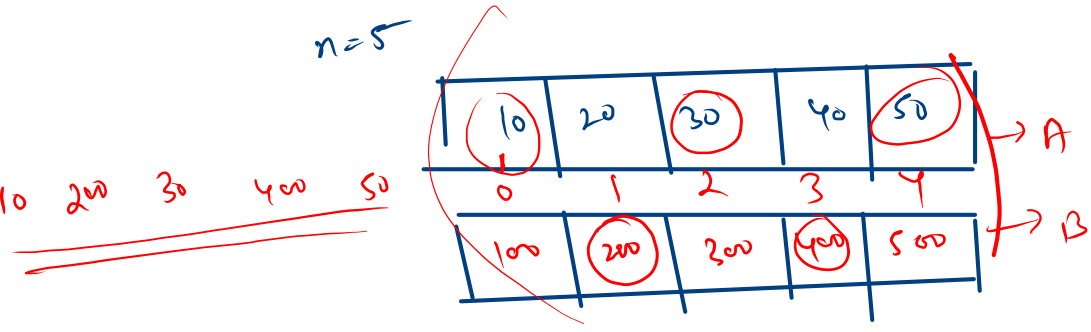
Print the first element of the first array present at the **0th** index, then the element of the **second array** at the **1st** index, then the element of the first array at the **2nd** index, then the element of the second array at the **3rd** index, so on and so forth.

**Sample Input 0**

```
5
10
20
30
40
50
100
200
300
400
500
```

**Sample Output 0**

```
10 200 30 400 50
```



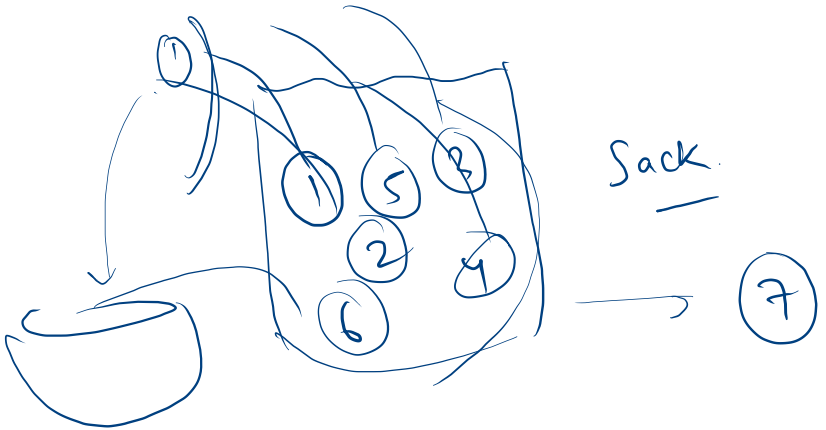

idx → even → A

idx → odd → B.

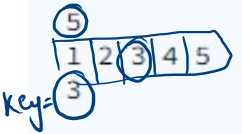
# Check if x is present in array or not

Problem | Submissions | Leaderboard | Discussions

Given an **array**, the task is to write a Java program to check whether a specific element is **present** in this Array or not.

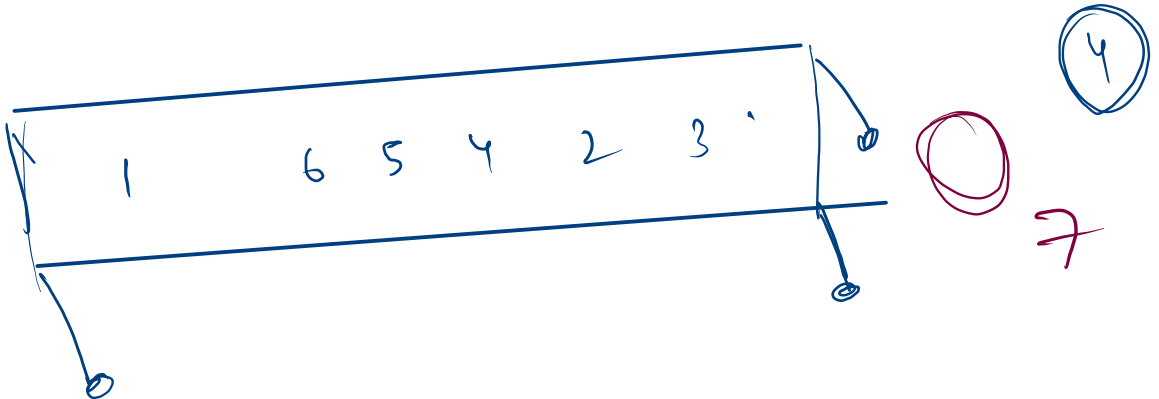


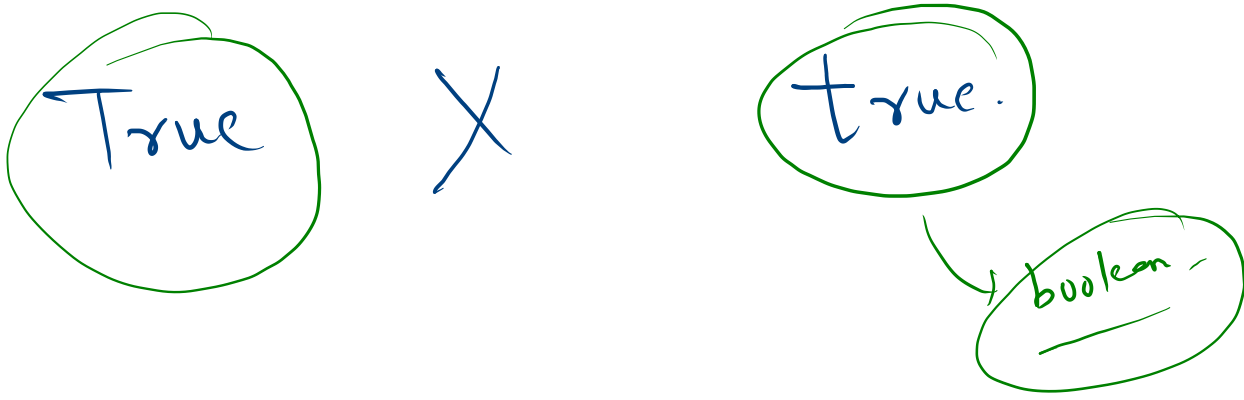

## Sample Input 0




## Sample Output 0

True




True
X
true.
boolean

```
3
4  public class Solution {
5      public static String search(int [] A, int key){
6          for(int i = 0 ; i < A.length; i++){
7              if(A[i] == key){
8                  return "True";
9              }
10         }
11         return "False";
12     }
13
```