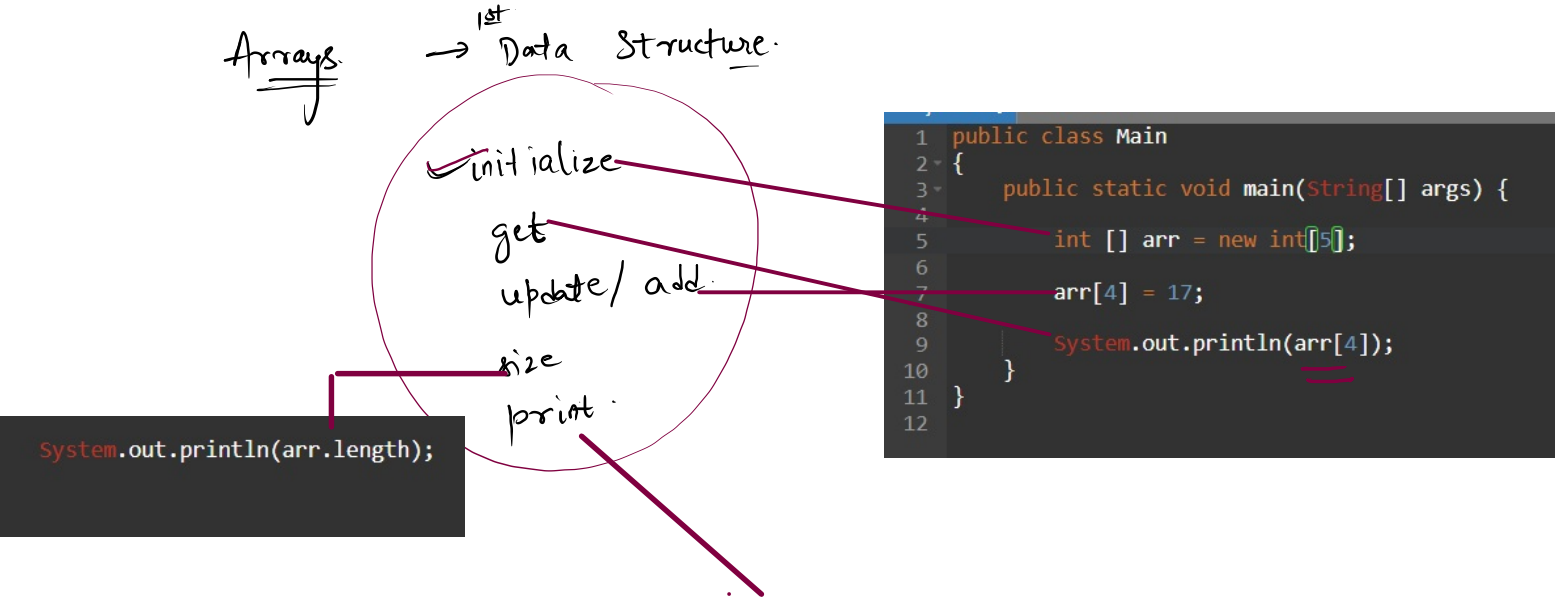

Arrays → 1st Data Structure.
initialize
get
update / add
size
print
System.out.println(arr.length);
1 public class Main
2 {
3 public static void main(String[] args) {
5 int [] arr = new int[5];
7 arr[4] = 17;
9 System.out.println(arr[4]);
10 }
11 }

8 students.

Wrong Question.

↳ +2

=

```
int marks0 = 18  20
int marks1 = 20  22
int marks2 = 16  18
int marks3 = 13  14
int marks4 = 15  17
int marks5 = 20  22
int marks6 = 13  15
int marks7 = 17  19
```

8 variables.

500 students.

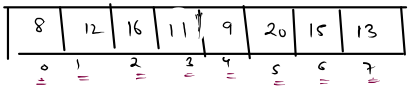
marks.

Array { group of similar kind of data stored at contiguous location.

(same data-type)

8
12
16
17
9
20
15
13

marks



"aman"
0 1 2 3.

len = 8

[0, len-1]

allocate some memory.

```
int [] arr = new int[8];
```

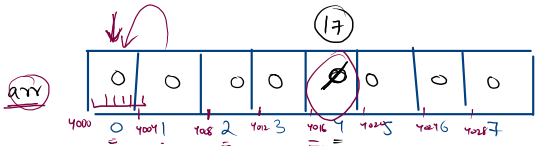
default value of int in Java.

name provided by you.

size of array.

data-type

what to store?




```
1  public class Main
2  {
3      public static void main(String[] args) {
4
5          int [] arr = new int[8];
6
7          arr[4] = 17;
8
9          System.out.println(arr[4]);
10     }
11 }
12
```

update

get

arr[4] = 17

L = R

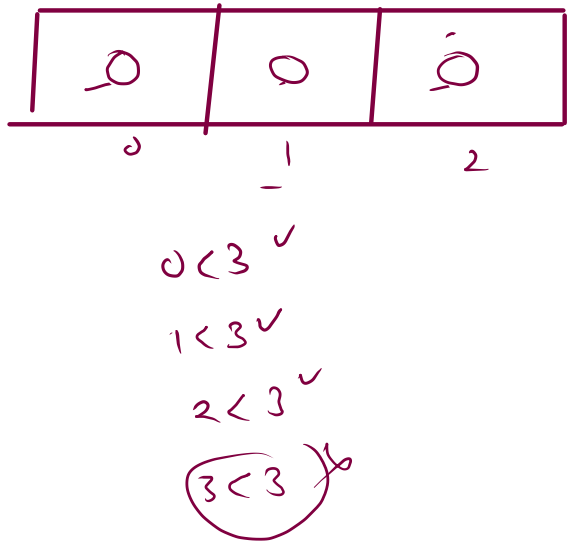
12th

arr[4] ⇒ 4000 + 4 × (size of int)

4000 + (4×4)

= 4016

```java
int [] arr = new int[3];

arr[0] = 10;
arr[1] = 20;
arr[2] = 30;

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



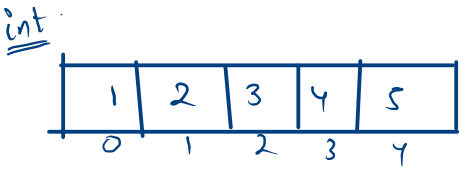
Print the array elements linewise

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.

And print each integer in each line.



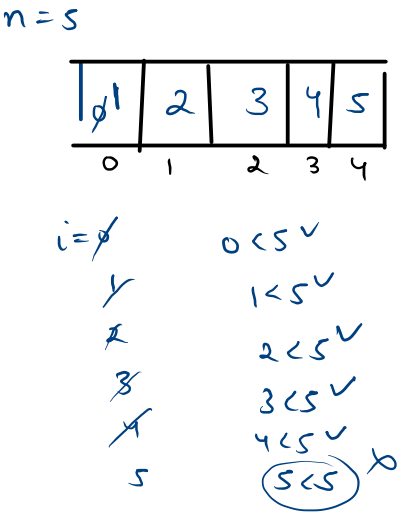
Sample Input 0

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

Sample Output 0

```
1
2
3
4
5
```

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

        //output
        for(int i = 0; i < n; i++){
            System.out.println(A[i]);
        }

    }
}
```

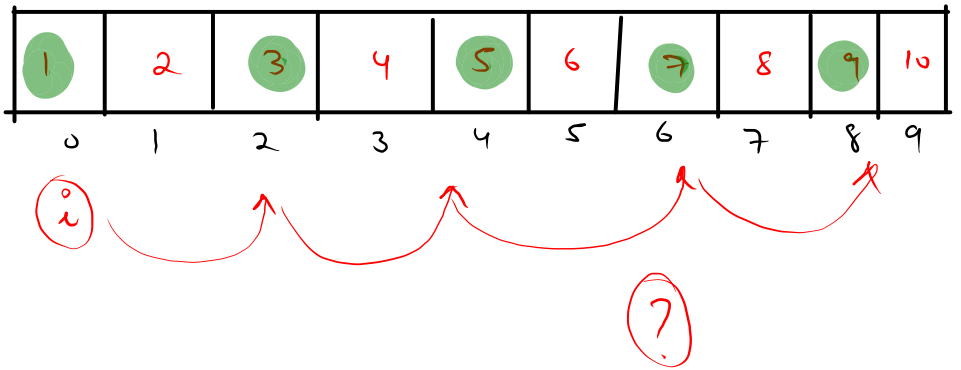


Print Alternate Array Elements Linewise

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



Sample Input 0

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

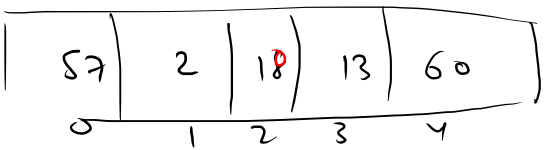
Sample Output 0

```
1
3
5
7
9
```

eg.

5 n

57
2
18
13
60



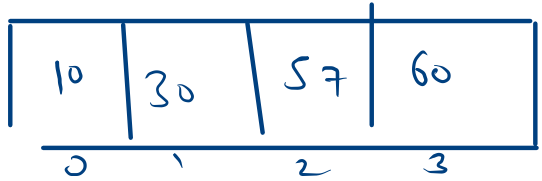
57 18 60

```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];
        //input for array
        for(int i = 0 ; i < n; i++){
            A[i] = scn.nextInt();
        }
        //output
        for(int i = 0; i < n; i += 2){
            System.out.println(A[i]);
        }


    }
}
```

4

10 30 57 60

n = 4



i = 0 → 2 → 4

0 < 4 .

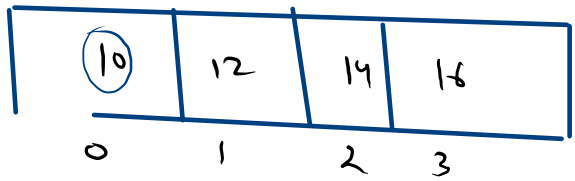
2 < 4

4 < 4 ✗

10
57

```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();


        }
        for(int i=0;i<n;i++)
            if(a[i]%2!=0)
                System.out.println(a[i]);
    }
}
```

n=4



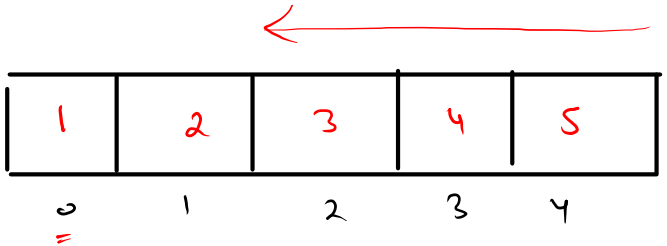
i=0 0<4 ✓

10%2 !=0 → odd.

i==0

!=0
i==0 even.

Print Array Elements Reverse linewise



reverse loop.

last Idx = n-1

i = last Idx ⟶ 0

A[i]

4 3 2 1 0

A[i]

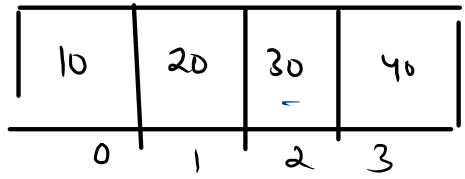
Sample Input 0

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

Sample Output 0

```
5 4 3 2 1
```

n = 4




```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();
        }

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

    }
}
```

i = 3, 2, 1, 0, -1

3 ≥ 0 ✓
2 ≥ 0 ✓
1 ≥ 0 ✓
0 ≥ 0 ✓
-1 ≥ 0 ✗

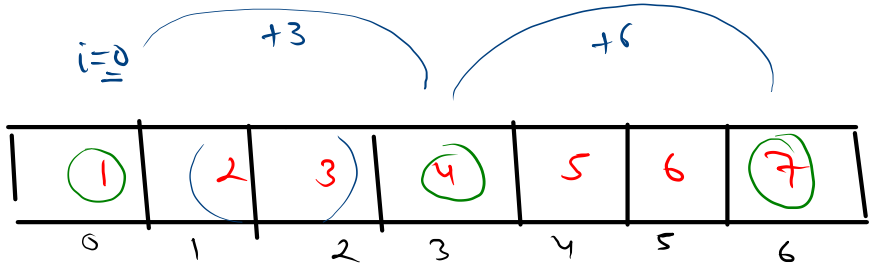
40 30 20 10

Print Array element if index divisible by 3

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

Sample Output 0

```
1 4 7
```



if

i%3==0

0%3==0

1%3!=0

2%3!=0

3%3==0

4%3!=0

5%3!=0

6%3==0

n=7

1 2 3 4 5 6 7

0 1 2 3 4 5 6

i=0 0<7 ✓

0%3==0 ✓

1 1<7 ✓

1%3==0

2 2<n 2<7 ✓

2%3==0

i=3 3%3==0 ✓

```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();
        }
        for(int i = 0; i < n; i++){
            if(i % 3 == 0){
                System.out.print(A[i] + " ");
            }
        }
    }
}
```

Check if two arrays are identical?

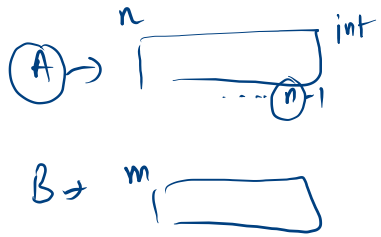
Problem | Submissions | Leaderboard | Discussions

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



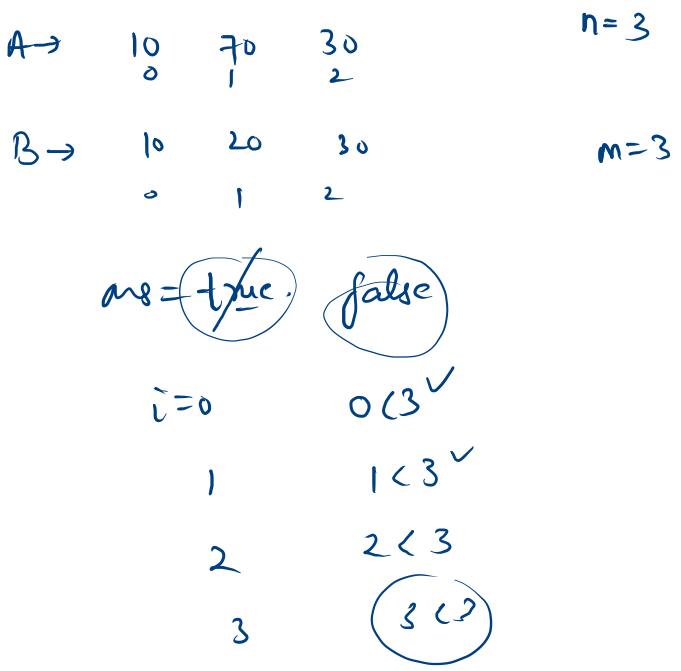
Sample Input 0

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

Sample Output 0

```
true
```

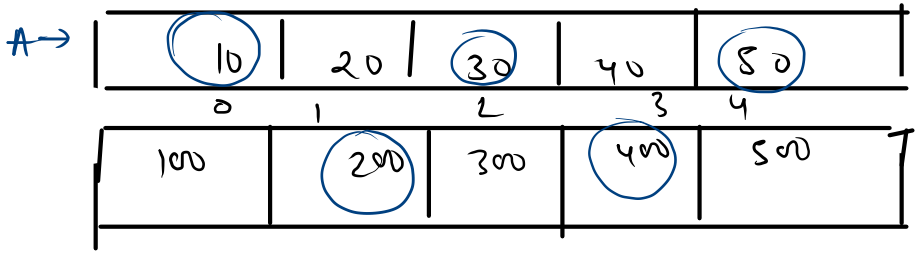
```java
    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int [] A = new int[n];
        for(int i = 0; i < n; i++){
            A[i] = scn.nextInt();
        }
        int m = scn.nextInt();
        int [] B = new int[m];
        for(int i = 0; i < m; i++){
            B[i] = scn.nextInt();
        }
        boolean ans = true;     //assume they are identical
        if(n != m){
            ans = false;
        }
        else{   //n == m we need to check elements
            for(int i = 0; i < n; i++){
                if(A[i] != B[i]){
                    ans = false;
                }
            }
        }
        System.out.println(ans);
    }
}
```



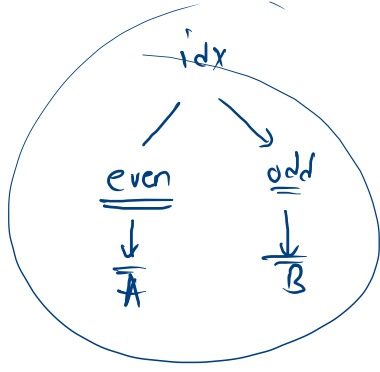
Print two arrays alternately

Sample Input 0

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



10 200 30 400 50



Sample Output 0

```
10 200 30 400 50
```