

Handwritten diagram illustrating memory access and loop iteration:

- Array:** `arr` with values `10, 20, 30, 40, 50` and indices `0, 1, 2, 3, 4`.
- Memory Address:** `arr` points to address `4K`.
- Calculation:** `arr[2] = 30` is calculated as `4000 + 2 * 4 = 4008`.
- Loop Condition:** `i <= 5` is shown, with a calculation `5 < 5` and `i = 5`.
- Loop Body:** `for (int i = 0; i <= arr.length; i++)` with `int val = arr[i]`.

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.

```

N
int[] arr = new int[N]
for (int i = 0; i < arr.length; i++) {
    arr[i] = sup.nextInt();
}

```

$\begin{array}{|c|c|c|c|c|} \hline 10 & 20 & 30 & 40 & 50 \\ \hline \end{array}$

(a) $\begin{array}{c} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{array}$

50
40
30
20
10

```
public class Solution {
    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<arr.length; i++){
            arr[i] = scn.nextInt();
        }
        for(int i= arr.length-1; i>=0; i--){
            System.out.print(arr[i]+" ");
        }
    }
}
```

$\begin{array}{r} 1 \\ 0 \\ + \\ 2 \\ \hline 2 \end{array}$

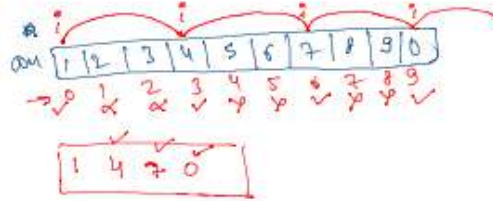
2004
 2004

Q2

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

Example:

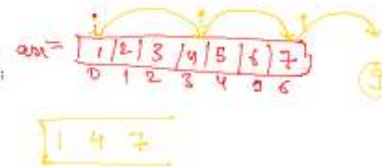


```
public class Solution {
    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int[] arr = new int[n];

        for(int i=0; i<arr.length; i++){
            arr[i] = scn.nextInt();
        }

        for(int i=0; i<arr.length; i+=3){
            System.out.print(arr[i]+" ");
        }

        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class */
    }
}
```



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.



```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr1 = new int[n];

    for(int i=0; i<arr1.length; i++){
        arr1[i] = scn.nextInt();
    }

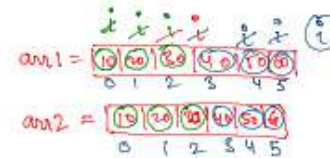
    // Scanner scn = new Scanner(System.in);
    int m = scn.nextInt();
    int[] arr2 = new int[m];

    for(int i=0; i<arr2.length; i++){
        arr2[i] = scn.nextInt();
    }

    if(arr1.length != arr2.length){
        System.out.println("false");
        return;
    }

    for(int i=0; i<n; i++){
        int val1 = arr1[i];
        int val2 = arr2[i];

        if(val1 != val2){
            System.out.println("false");
            return;
        }
    }
}
```



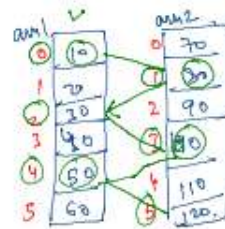
false / true

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.



$[10 \ 80 \ 30 \ 100 \ 50 \ 120]$
 $i, j, 2 == 0$
 $arr1[i];$
 $arr2[j];$

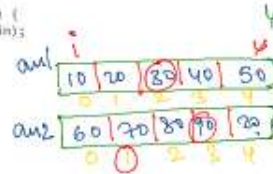
```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr1 = new int[n];

    for (int i=0; i<arr1.length; i++){
        arr1[i] = scn.nextInt();
    }

    int[] arr2 = new int[n];

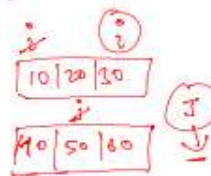
    for (int i=0; i<arr2.length; i++){
        arr2[i] = scn.nextInt();
    }

    for (int i=0; i<arr1.length; i++){
        if (i%2==0){
            System.out.print(arr1[i]+" ");
        } else {
            System.out.print(arr2[i]+" ");
        }
    }
}
```

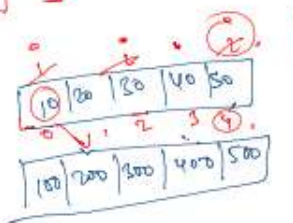


$i=0$ 10
 $i=1$ 20
 $i=2$ 30
 $i=3$ 40
 $i=4$ 50

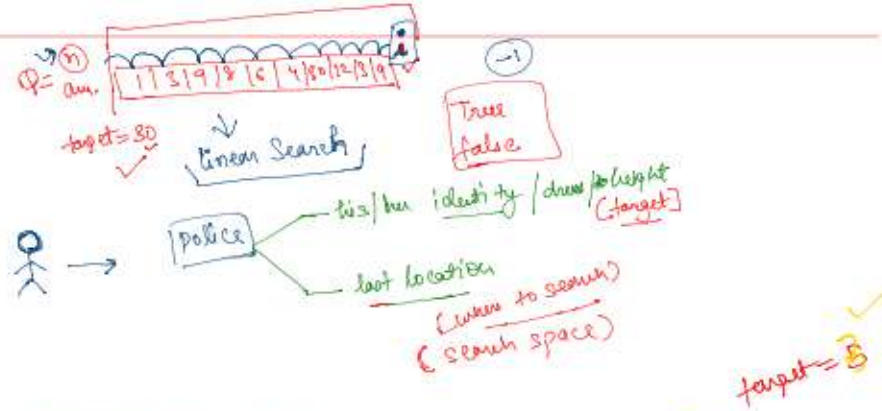
$for (int i=0, j=1; i<arr1.length; i+=2, j+=2){$
 $\quad sys (arr1[i]);$
 $\quad sys (arr2[j]);$



$for (int i=0; i<arr1.length; i+=2){$
 $\quad sys (arr1[i]);$
 $\quad if (i+1 < n){$
 $\quad \quad sys (arr2[i+1]);$



10
 20
 30
 40
 50




```
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<arr.length; i++){
        arr[i] = scn.nextInt();
    }

    int target = scn.nextInt();

    for(int i=0; i<arr.length; i++){
        if(arr[i]==target){
            System.out.println("True");
            return;
        }
    }

    System.out.println("False");
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. You */
}
```

am. 

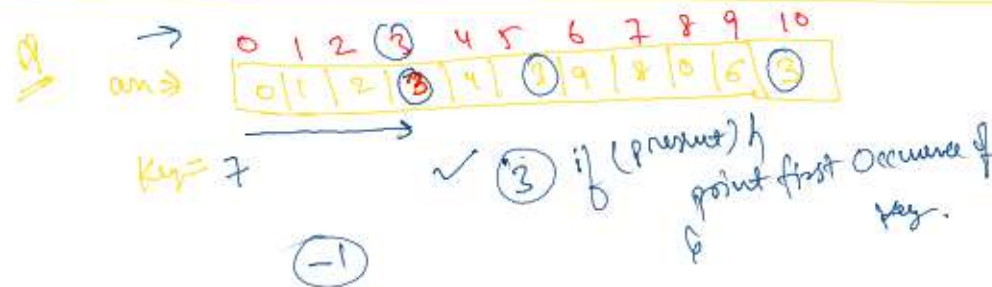
target = 30 ✓

True false

police

his/her identity / dress / height (target)

last location (where to search) (search space)



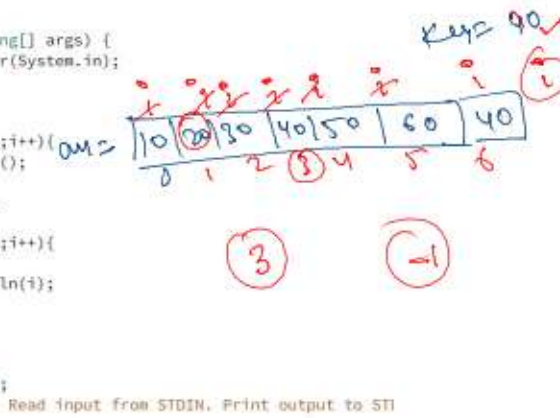
```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];

    for(int i=0; i<arr.length; i++){
        arr[i] = scn.nextInt();
    }

    int key = scn.nextInt();

    for(int i=0; i<arr.length; i++){
        if(arr[i]==key){
            System.out.println(i);
            return;
        }
    }

    System.out.println("-1");
    /* Enter your code here. Read input from STDIN. Print output to STI */
}
```

am. 

key = 40 ✓

True false

police

his/her identity / dress / height (target)

last location (where to search) (search space)