Arrays
& ArrayLists

What is a Data Structure?

Store data in a particular fashion

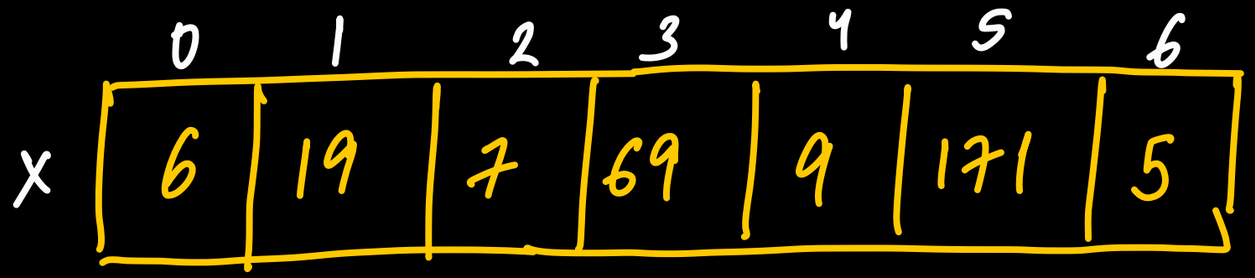
Variables → data ko store kar rahe the.

int cecilia = 100;

- Arrays
- Linked List
- Stack
- Queue
- Tree
- Graph
- Hashsets

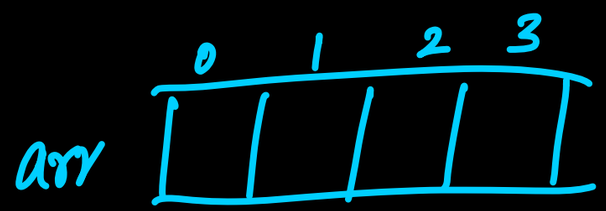
What is a Array?

Linear Data Structure.



Initialisation & Indexing

```
int[] arr = new int[4];
```



Output & Input of Array using Loop

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

```
for(int i=0; i<n; i++){
    arr[i] = sc.nextInt();
}
```

Ques: Given an Array, print negative elements only

```
int arr[] = {                    };
```

Default Values

Ques: Print sum of elements of the array.

```
int[] arr = {-6, 8, 14, -2, 23, 47};
int sum = 0;
for(      ){
|   sum += arr[i];
}
Sout(sum);
```

HW: Print product of elements of the array.

Ques: Print the Maximum element in the array [Largest element in Array]

i

```
int[] arr = {-6, 8, 14, -2, 23, 47, 4, 3, 10}

int max = arr[0];

for(int i=0; i<n; i++){
|     if(arr[i] > max)  max= arr[i];
3
sout(max);
```

max= -6 8 14 23 47

HW: Print the minimum element in the array

Array Index of out bound Exception

access, print, input le galat indices ko to error aata hai

n → 0 to n-1

arr[-1] → Error

Array of Other Data Types

```
int[] arr = {10, 20, 30}
```

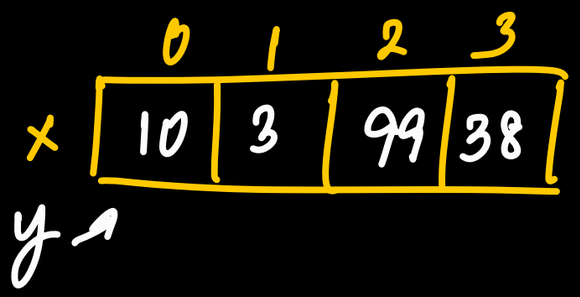
or

```
int[] brr = new int[100];
```

int[] x

Passing Array to Methods

```java
public static void main(String[] args) {
    int x[] = {10,3,29,38};
    System.out.println(x[2]);
    change(x);
    System.out.println(x[2]);
}
public static void change(int[] y) {
    y[2] = 99;
}
```

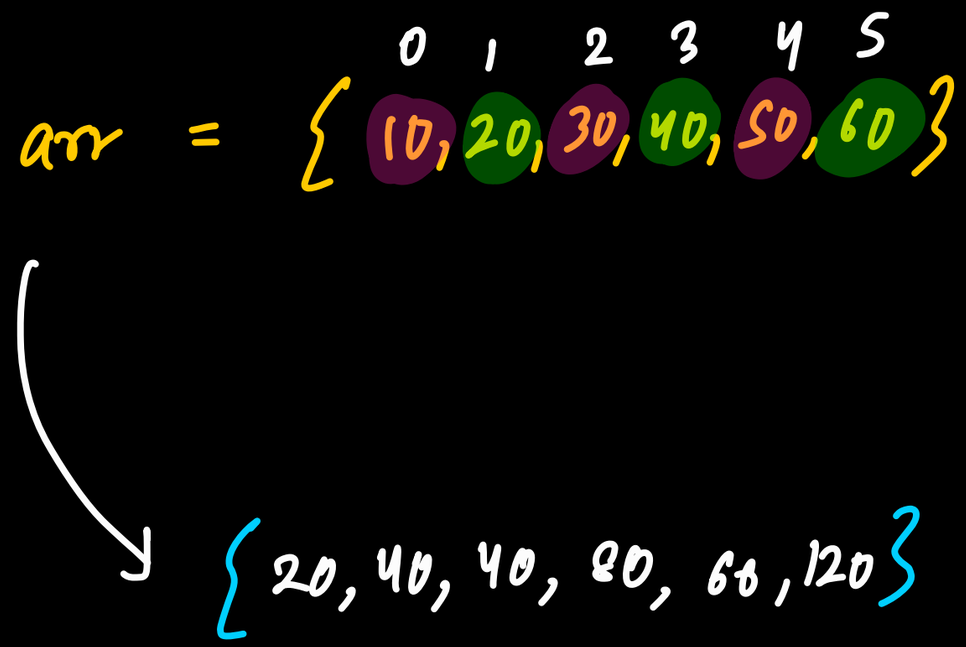


Output

29

99

HW : Multiply odd indexed elements by 2 and add 10 to even indexed elements



HW: Print product of elements of the array.

HW: Print the minimum element in the array

Ques: Multiply odd indexed elements by 2 and add 10 to even indexed elements

Ques: Search in Array

Ques: Two Sum

arr = {1, 5, 8, -3} target = 2

```
for(int i=0; i<n; i++){
    for (int j=i+1; j<n; j++){
        if(arr[i] + arr[j] == target) —
    }
}
```

Ques: Two Sum

```
          0  1  2  3  4  5
arr = { 1, 4, 45, 6, 10, 8 }      tar = 16      n = 6
                  i  j
```

```
for(int i=0; i<n; i++){
    for (int j=i+1; j<n; j++){
        if(arr[i] + arr[j] == target) return true
    }
}
return false;
```

Ques: Print the second Maximum element in the array

arr = { 4, 10, 10, 6, 3, 8 }

max = 10

Smax = 8

Approach : Find largest element (max)

Now find largest element except max

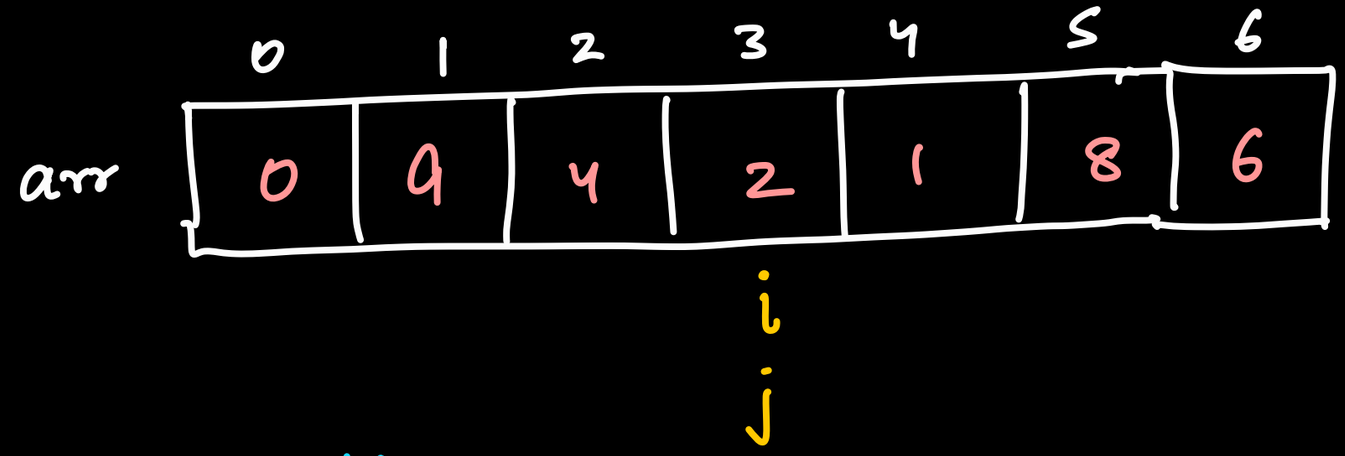
For Each Loop

used to traverse data structures

cannot modify array elements

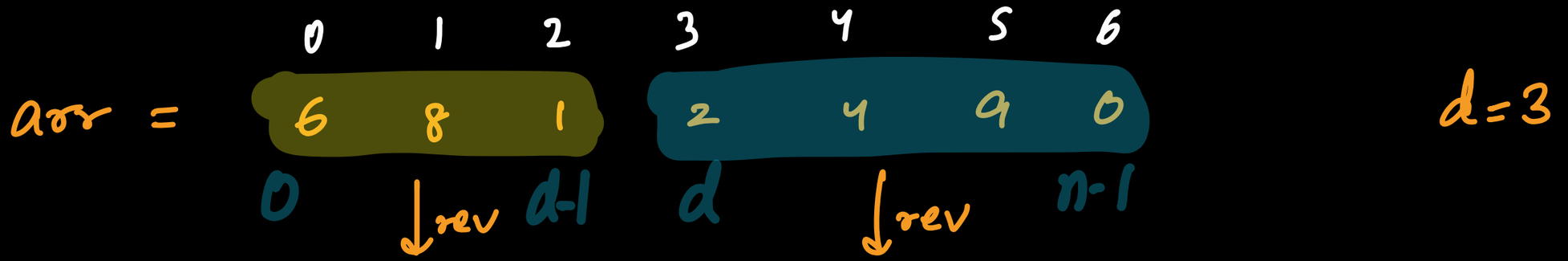
Ques: Reverse Array (2 variable/pointer technique)

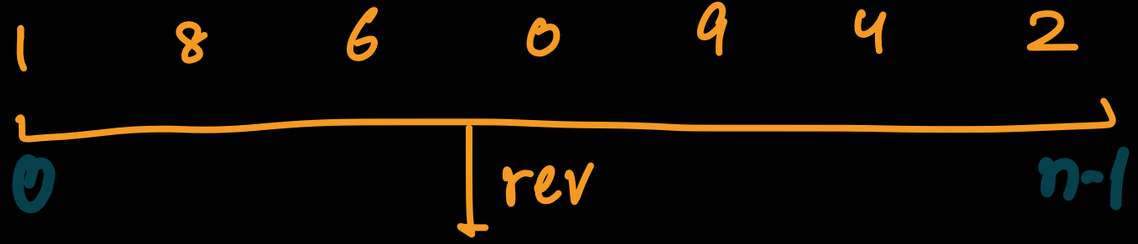
arr = { 6, 8, 1, 2, 4, 9, 0 }




```
while(i<j)
{
    swap(arr[i], arr[j])
    i++;
    j--;
}
```

* Ques: Rotate Array





2 4 9 0 6 8 1

Ques: Rotate Array

arr = 1 2 3 4

d = 6

↓

d = 2

3 4 1 2

d = d % n

Ques: Missing in Array

Ques: Segregate 0s and 1s

arr = { 1, 0, 0, 1, 1, 1, 0, 1, 0 1, 1, 0 }

M-1: Count the no. of zeroes & ones

noZ = 5, noO = 7

```
for (int i = 0; i < noZ ; i++){
|   arr[i] = 0
3
for (int i = noZ ; i<n; i++){
|   arr[i] = 1;
3
```

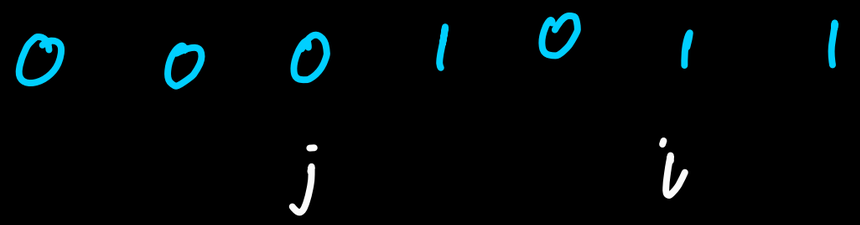
Ques: Segregate 0s and 1s (2-pointer technique)

arr = { 1, 0, 0, 1, 1, 1, 0, 1, 0 1, 1, 0 }

0 0 0 0 0 1 1 1 1 1 1 1

j i

Ques: Segregate 0s and 1s




```
while(i<j) {
    if(arr[i]==0) i++;
    if(arr[j]==1) j--;
    if(arr[i]==1 && arr[j]==0) {
        swap
        i++
        j--
    }
}
```

Ques: Wave Array

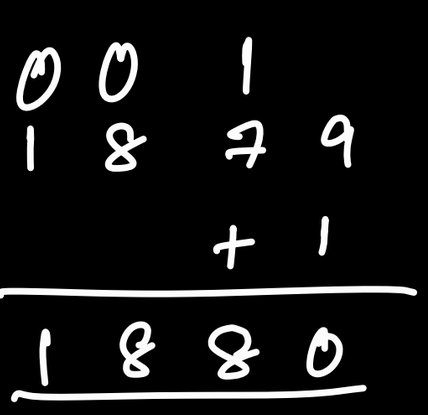
Arraylist & Vector in Java

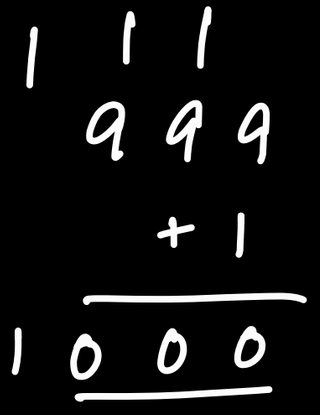
dynamic arrays → growing array

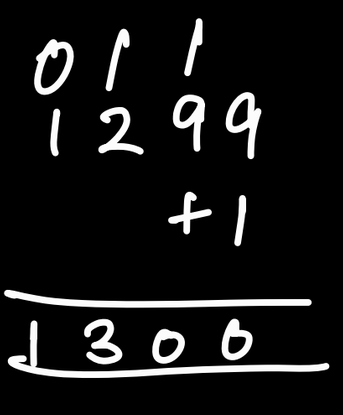
'Collection Framework'

Array ki problem → fixed size

Ques: Adding One








```
if(arr[n-1] ≠ 9)  arr[n-1]++;
```

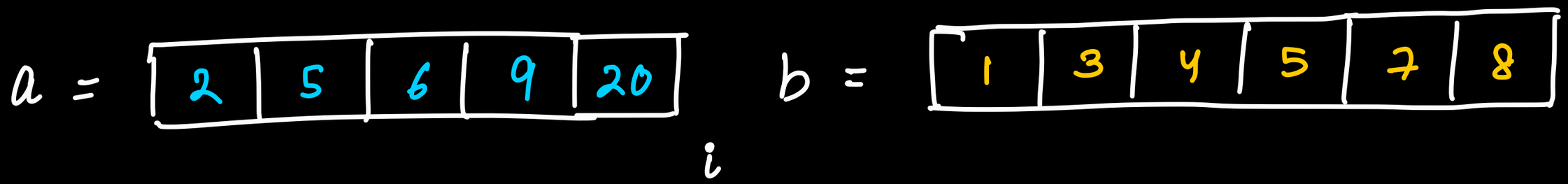
ans = {0, 8, 8, 1}

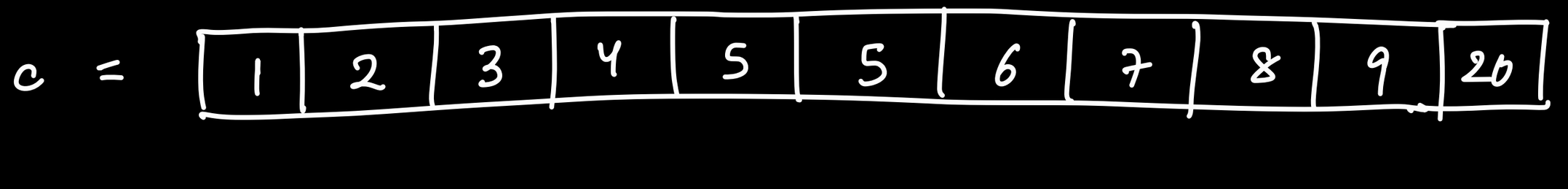
d

reverse

```
ArrayList<Integer> ans = new ArrayList<>();
```

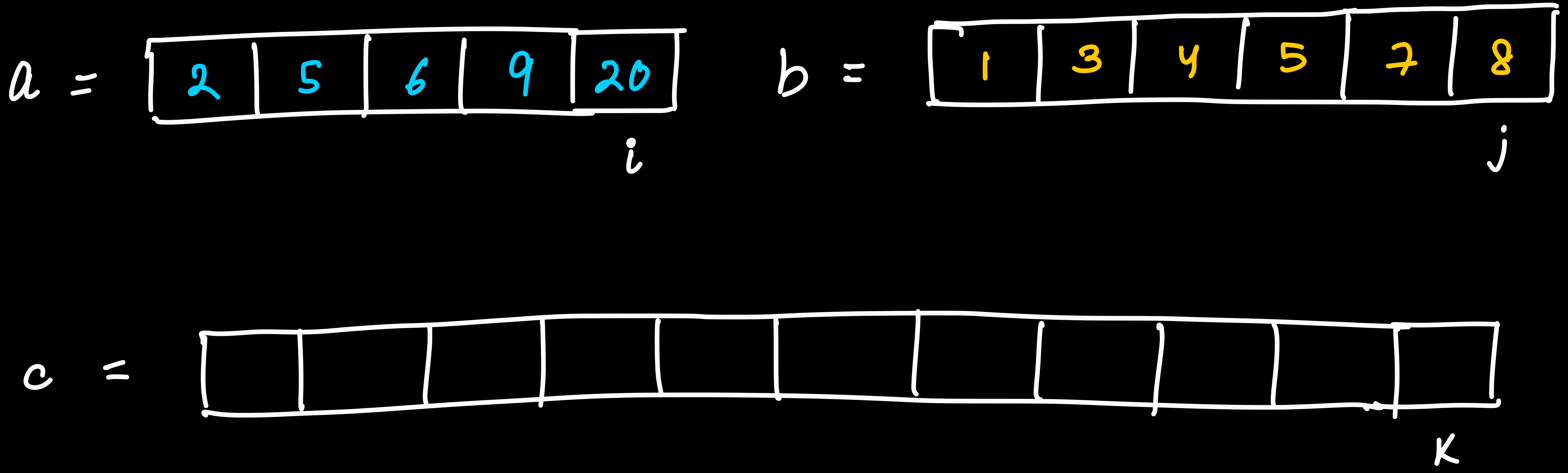
Ques: Merge 2 Sorted Arrays in a single big array.





3 pointer technique

Ques: Merge 2 Sorted Arrays 'Homework'



merge in reverse order

Ques: Sort 0s, 1s and 2s ('Homework')

THANKYOU

Cuties

THANKYOU

Cuties