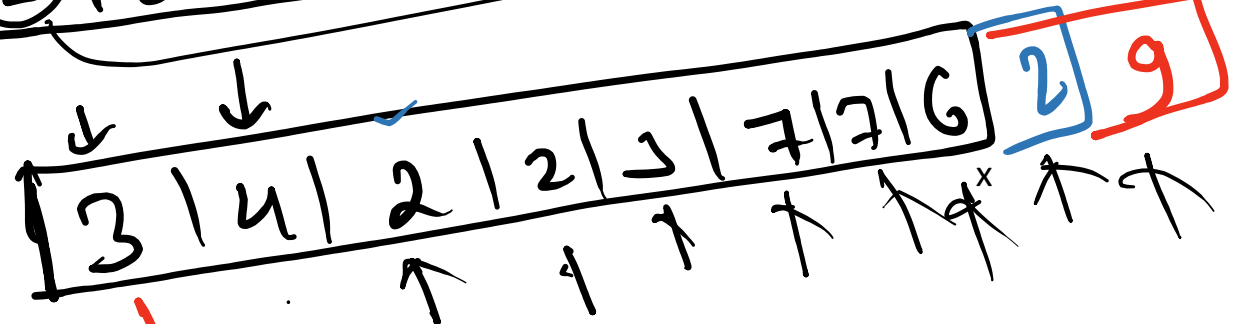
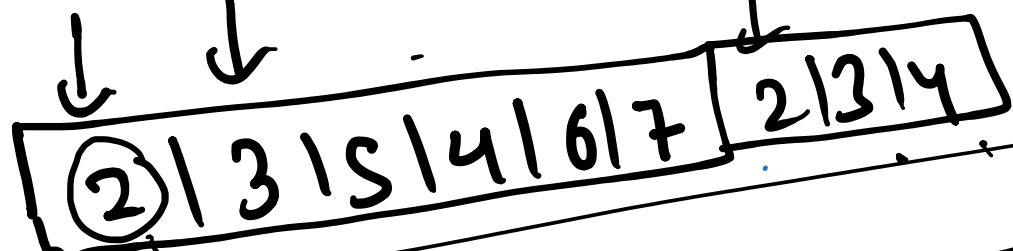


##

(i)

210

3 14, 2, 2, 13, 7, 6



2	0
3	0
5	1
4	1
6	0
7	0

```
public static int[] Intersection(int[] nums1, int[] nums2) {  
    HashMap<Integer, Integer> map = new HashMap<>();  
    for (int i = 0; i < nums1.length; i++) {  
        if (map.containsKey(nums1[i])) {  
            map.put(nums1[i], map.get(nums1[i]) + 1);  
        } else {  
            map.put(nums1[i], 1);  
        }  
    }  
}
```

nums2()

2	2	0
3	2	0
5	1	
4	2	1
6	1	
7	1	

3, 4, 2, 2, 7, 6

```
List<Integer> ll = new ArrayList<>();  
for (int i = 0; i < nums2.length; i++) {  
    if (map.containsKey(nums2[i]) && map.get(nums2[i]) > 0) {  
        ll.add(nums2[i]);  
    }  
}
```

ans = 2/5

2, 2, 5, 10, 4, 4, 3, 7, 14, 7, 8, 1, 15, 16, 20

2, 8, 10, 4, 1, 3, 7, 1, 8, 15, 16, 20

String builder

2x(Old cap) + 2

3, 4 = 7  
2, 2 = 9  
2, 2 = 10

2, 2, 3, 3, 4  
4, 3, 3, 4

[1, 3, 4, 2] -> size()  
[1, 7, 2]  
[1, 9]  
(10) ->

3, 3, 4  
6, 2  
10  
1, 2 = 3  
3, 2 = 6  
6, 4 = 10  
19

3, 3, 5, 5, 17  
6, 5, 5, 17  
6, 11, 6, 17

2, 2 = 4  
3, 3 = 6  
4, 4 = 8  
5, 5 = 10  
6, 6 = 12  
7, 7 = 14  
8, 8 = 16  
9, 9 = 18  
10, 10 = 20  
11, 11 = 22  
12, 12 = 24  
13, 13 = 26  
14, 14 = 28  
15, 15 = 30  
16, 16 = 32  
17, 17 = 34  
18, 18 = 36  
19, 19 = 38  
20, 20 = 40