

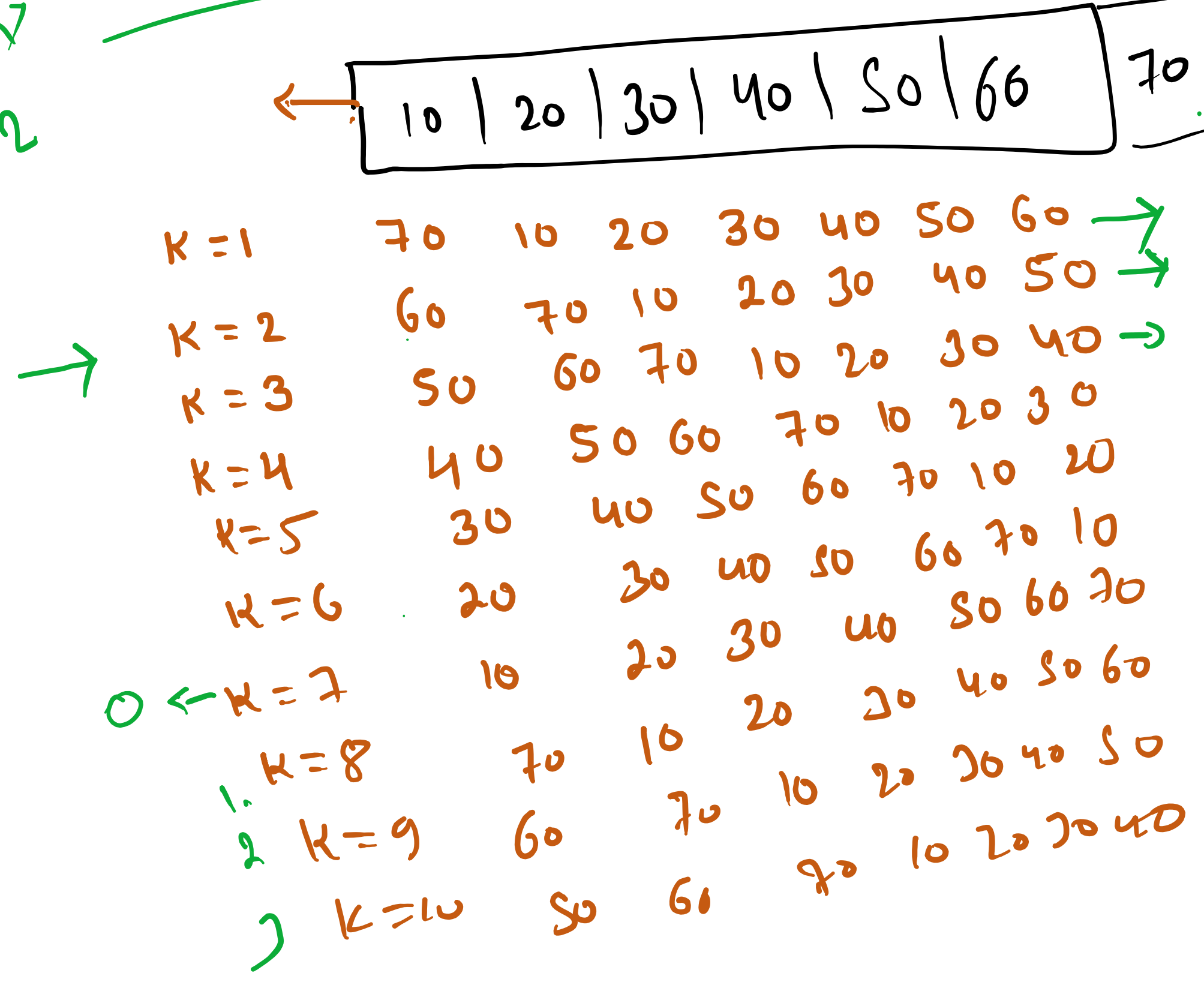
int  $\rightarrow -2^31 \rightarrow 2^31-1$

int[] arr = { 3, 5, 6, 2, 4, 16, 7, 8, 9 };

16

### Rotate Array

$K=142 \rightarrow 142 \% 7 = 2$



$K=3 \rightarrow \text{same time}$   
 $K = K \% N$   
 $8 \% 7 = 1$

70 10 20 30 40 50 60

item = arr[n-1]

arr[0] = item

i = n-1; i >= 0; i--  
arr[i+1] = arr[i]

70 10 20 30 40 50 60

K=3

i=5 arr[6]=arr[5]  
i=4 arr[5]=arr[4]  
i=3 arr[4]=arr[3]  
i=2 arr[3]=arr[2]  
i=1 arr[2]=arr[1]  
i=0 arr[1]=arr[0]

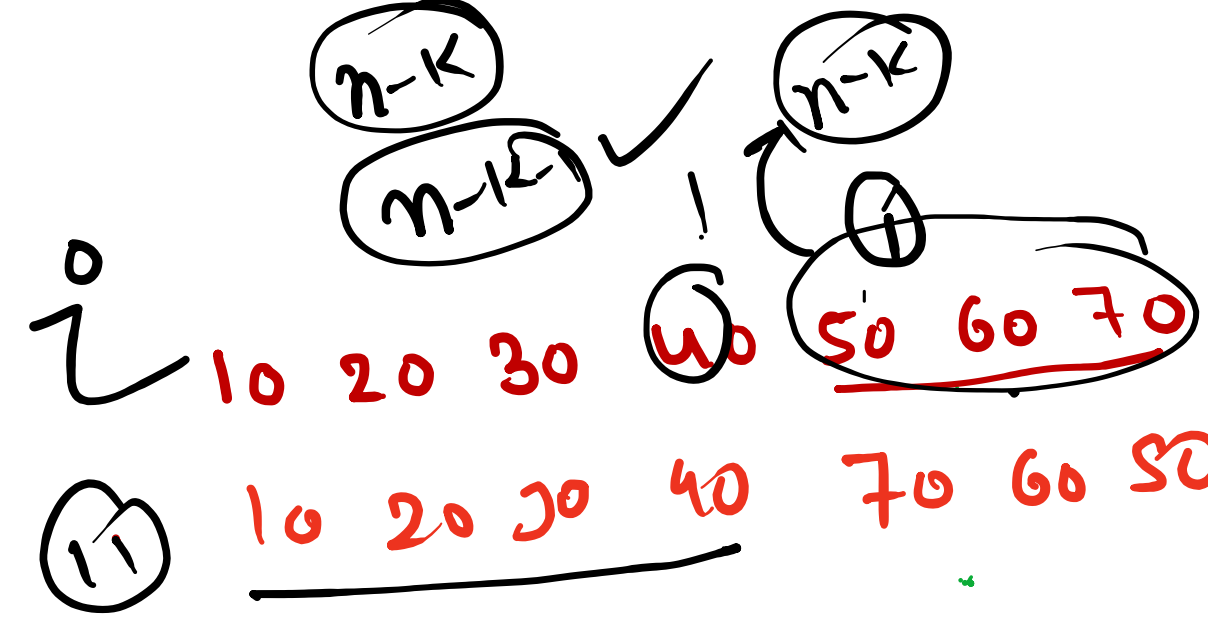
```
public static void Rotate(int[] arr, int k) {
    int n = arr.length;
    k = k % n;
    int item = arr[n-1];
    for (int i = n-2; i >= 0; i--) {
        arr[i+1] = arr[i];
    }
    arr[0] = item;
}
```

```
public static void Rotate(int[] arr, int k) {
    int n = arr.length;
    k = k % n;
    for (int j = 1; j <= k; j++) {
        int item = arr[n-1];
        for (int i = n-2; i >= 0; i--) {
            arr[i+1] = arr[i];
        }
        arr[0] = item;
    }
}
```

30 40 50 10 20

i=3 arr[4]=arr[3]  
i=2 arr[3]=arr[2]  
i=1 arr[2]=arr[1]  
i=0 arr[1]=arr[0]

### Reverse Array



50 60 70 10 20 30 40

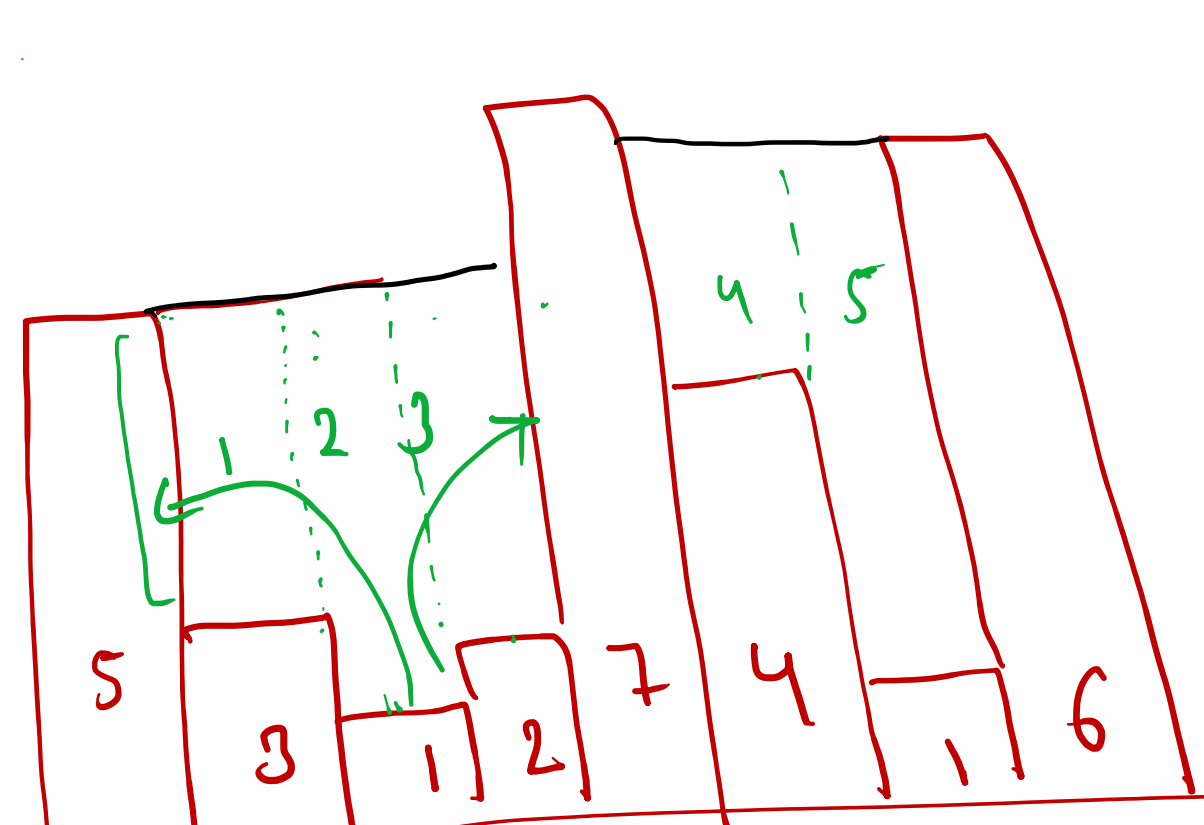
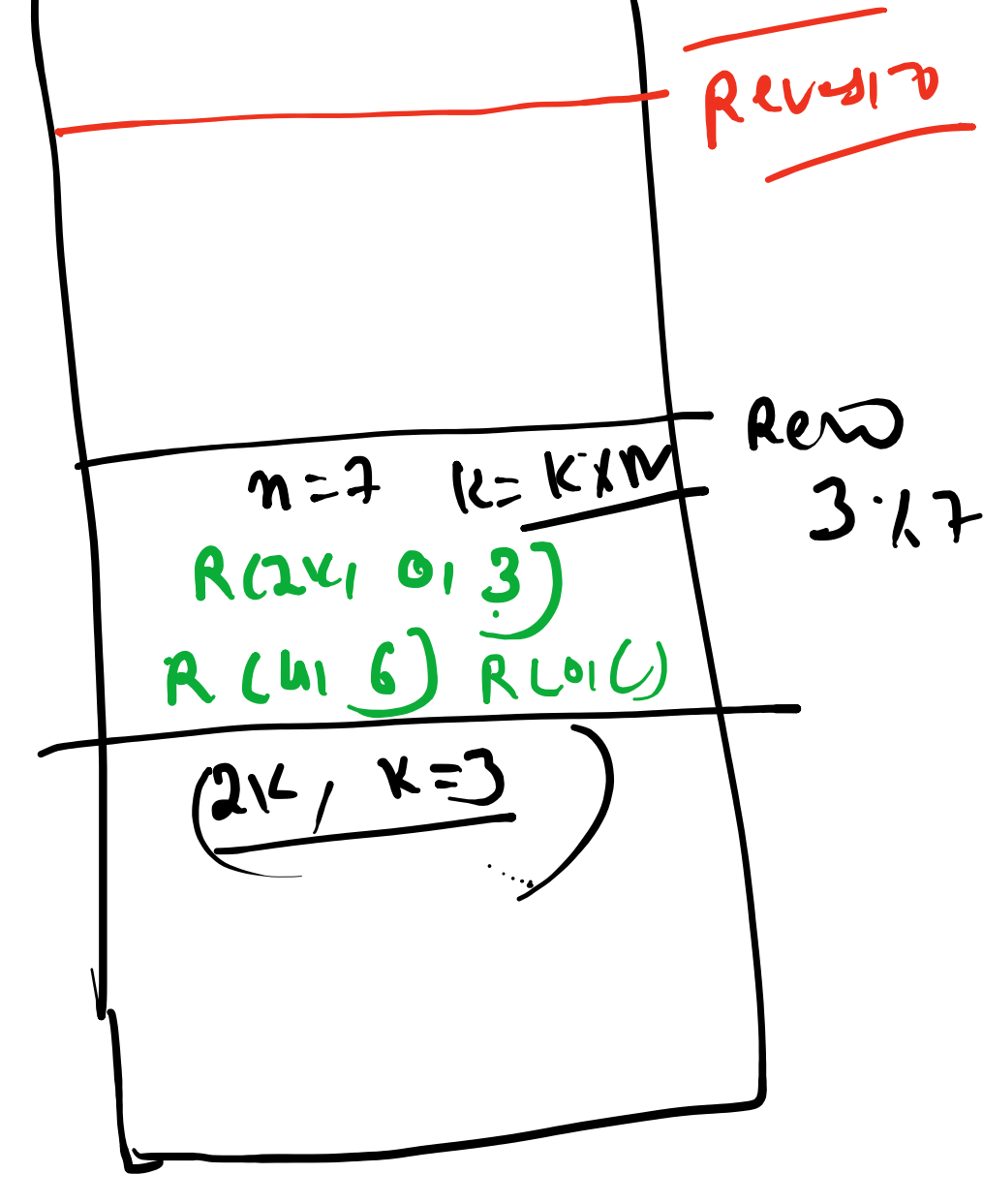
10 20 30 40 50 60 70

40 30 20 10 70 60 50

50 60 70 10 20 30 40

```
public static void Rotate(int[] arr, int k) {
    int n = arr.length;
    k = k % n;
    Reverse(arr, 0, n-k-1);
    Reverse(arr, n-k, n-1);
    Reverse(arr, 0, n-1);
}
```

```
public static void Reverse(int[] arr, int i, int j) {
    while (i < j) {
        int temp = arr[i];
        arr[i] = arr[j];
        arr[j] = temp;
        i++;
        j--;
    }
}
```



5, 3, 1, 2, 7, 4, 11, 6

min (Left[0], Right[0]) - arr[0] = (5, 7) - 5 = 0  
min (Left[1], Right[1]) - arr[1] = (5, 7) - 3 = 2  
min (Left[2], Right[2]) - arr[2] = (5, 7) - 1 = 4  
min (Left[3], Right[3]) - arr[3] = (5, 7) - 2 = 3  
min (Left[4], Right[4]) - arr[4] = (7, 7) - 7 = 0  
min (Left[5], Right[5]) - arr[5] = (7, 6) - 4 = 2  
min (Left[6], Right[6]) - arr[6] = (6, 6) - 11 = 5

Left: 5, 5, 5, 5, 7, 7, 7, 7  
Right: 7, 7, 7, 7, 6, 6, 6, 6

5, 3, 1, 2, 7, 4, 11, 6

Left Right Cal

7, 7, 7, 7, 6, 6, 6, 6

L[0] = arr[0]

5, 5, 5, 5, 7, 7, 7, 7

R[n-1] = arr[n-1]

i = n-2; i >= 0; i--

arr[i+1] = arr[i]

L[0] = max [L[0], arr[0]]  
L[1] = max [L[1], arr[1]]  
L[2] = max [L[2], arr[2]]  
L[3] = max [L[3], arr[3]]  
L[4] = max [L[4], arr[4]]  
L[5] = max [L[5], arr[5]]  
L[6] = max [L[6], arr[6]]  
L[7] = max [L[7], arr[7]]

R[6] = max [R[6], arr[6]]  
R[5] = max [R[5], arr[5]]  
R[4] = max [R[4], arr[4]]  
R[3] = max [R[3], arr[3]]  
R[2] = max [R[2], arr[2]]  
R[1] = max [R[1], arr[1]]  
R[0] = max [R[0], arr[0]]

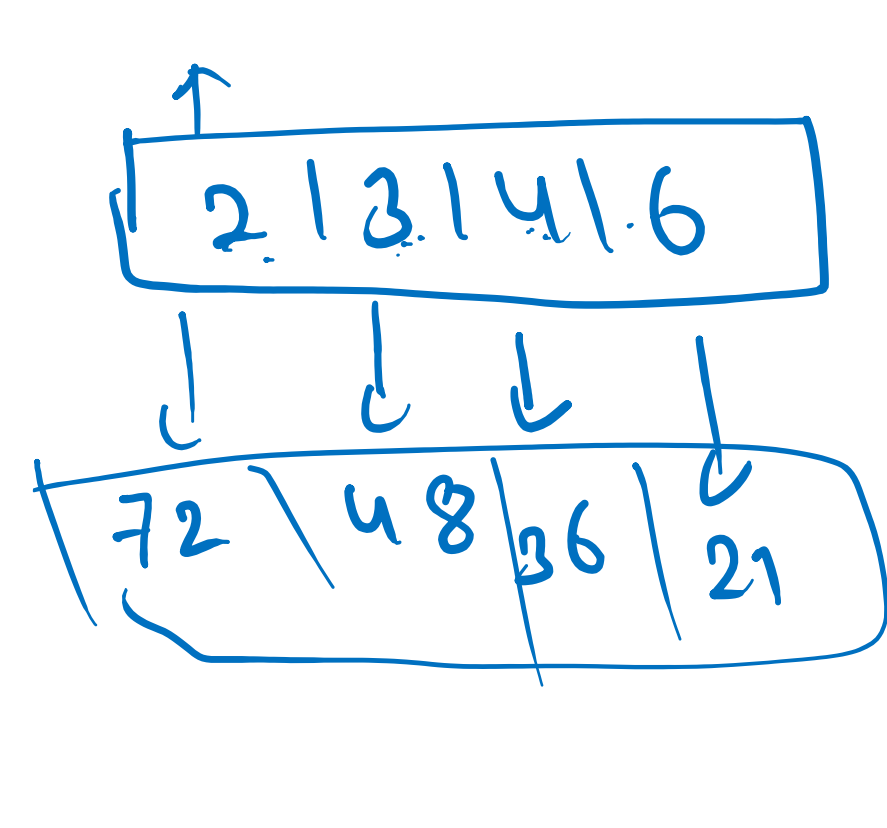
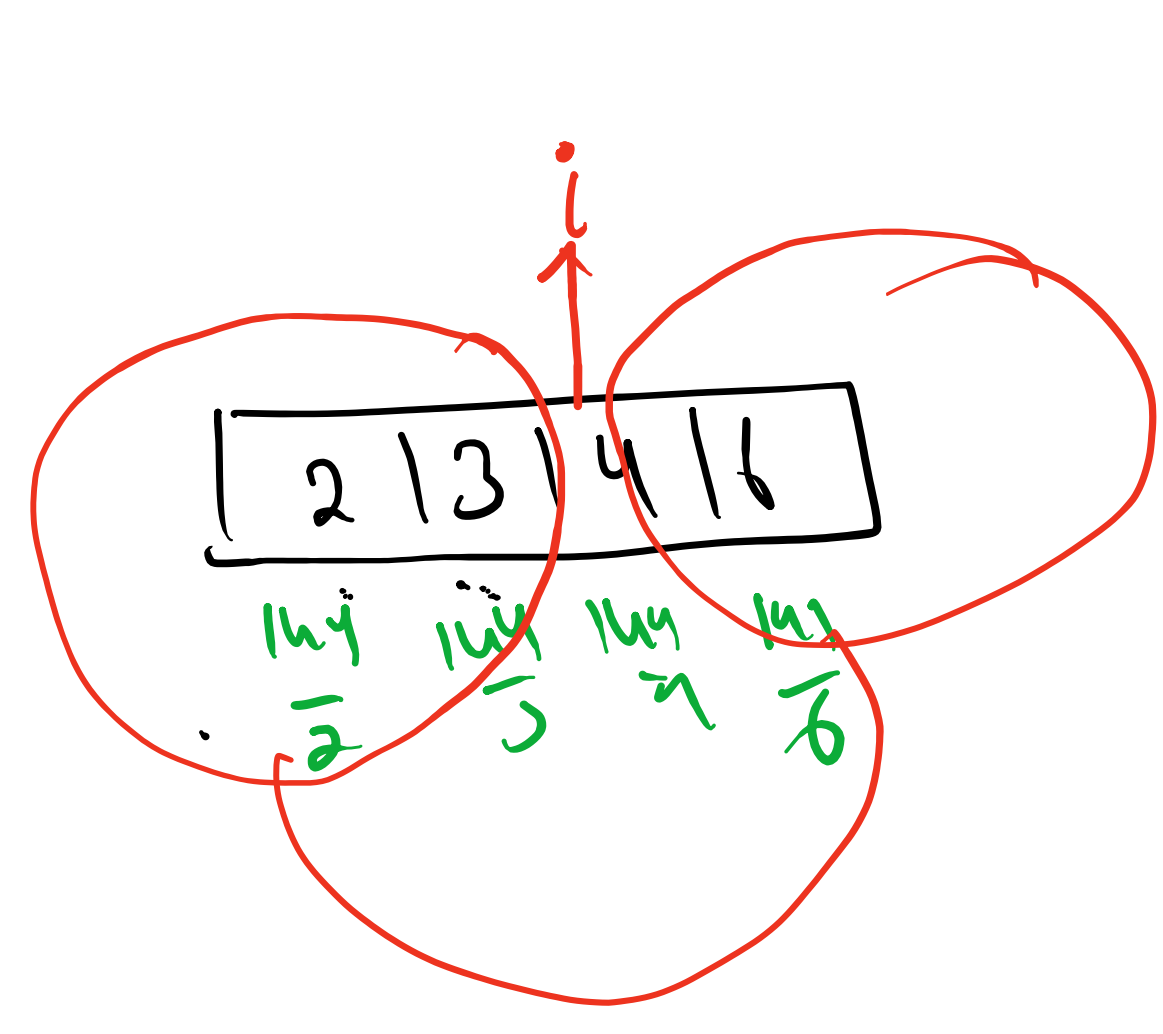
5, 3, 1, 2, 7, 4, 11, 6

min (L[0], R[0]) - arr[0] = (5, 7) - 5 = 0  
min (L[1], R[1]) - arr[1] = (5, 7) - 3 = 2

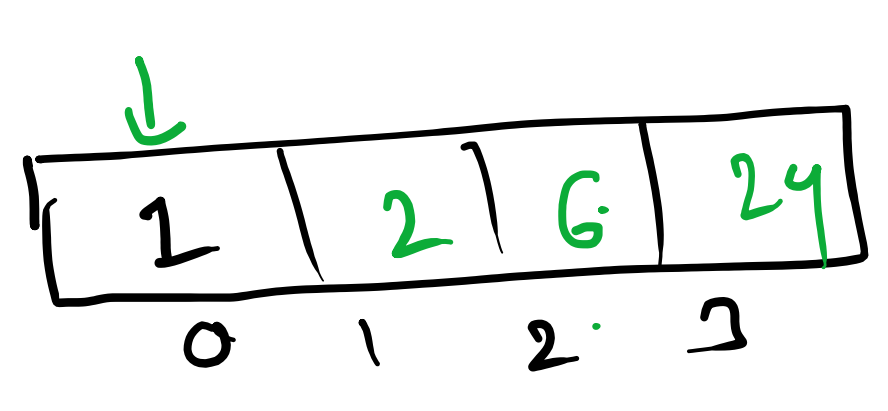
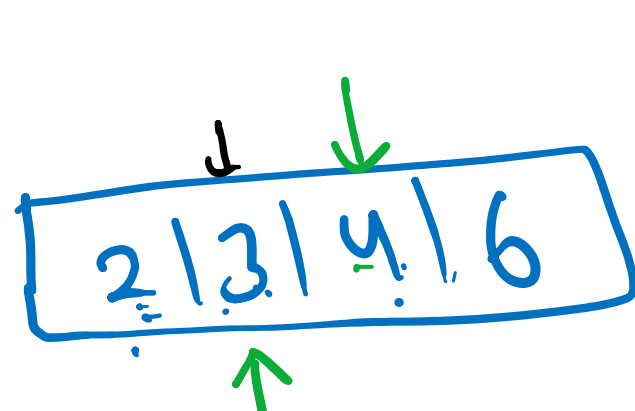
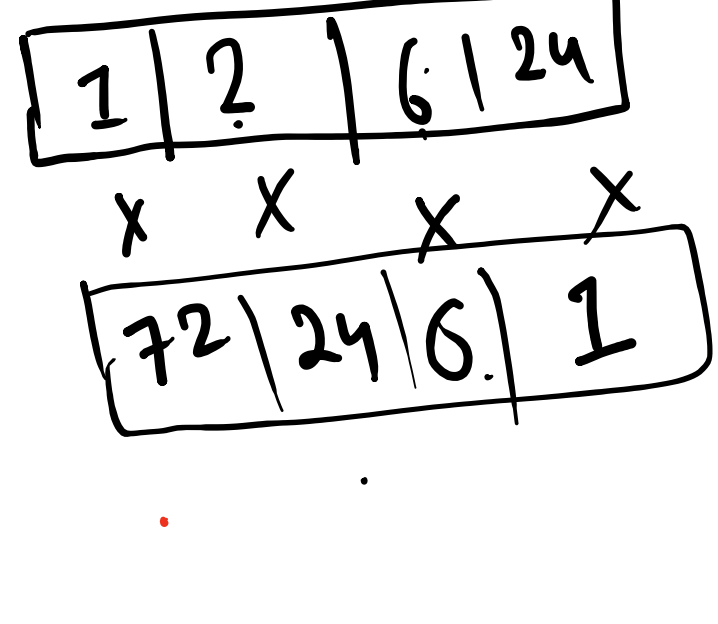
5, 5, 5, 5, 7, 7, 7, 7

min (L[4], R[4]) - arr[4] = (7, 7) - 7 = 0

7, 7, 7, 7, 6, 6, 6, 6



2x3x4x6 = 144



Left[i] = Left[i+1] \* arr[i+1]

Left[i] = Left[i] \* arr[i]

Left[i] = Left[i] \* arr[i]

Left[i] = Left[i] \* arr[i]