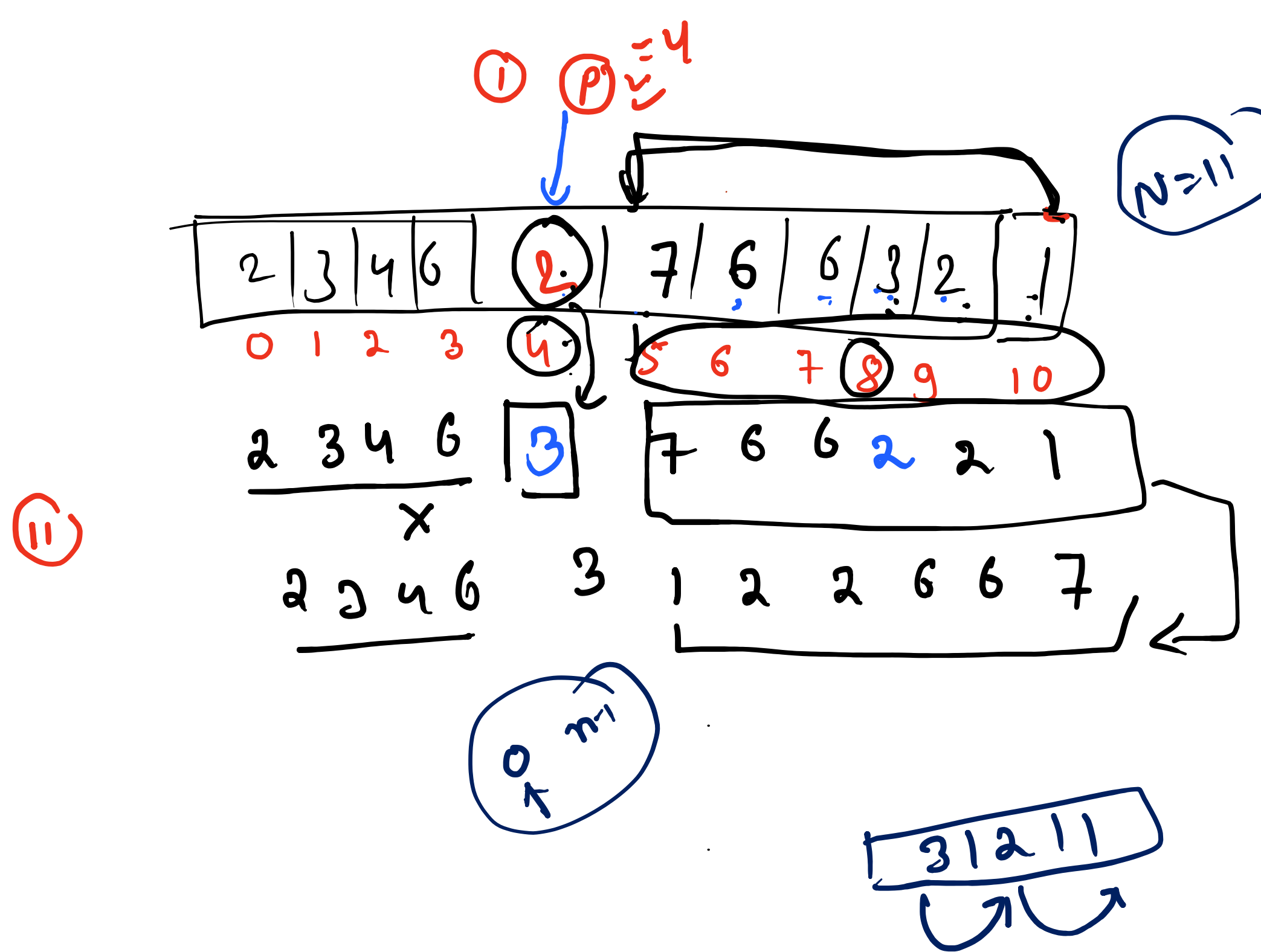


[3,2,1]
X
(1,2,3)

[1,2,3]
[1,3,2]

123 ✓
132 ✓
213
231
321
312



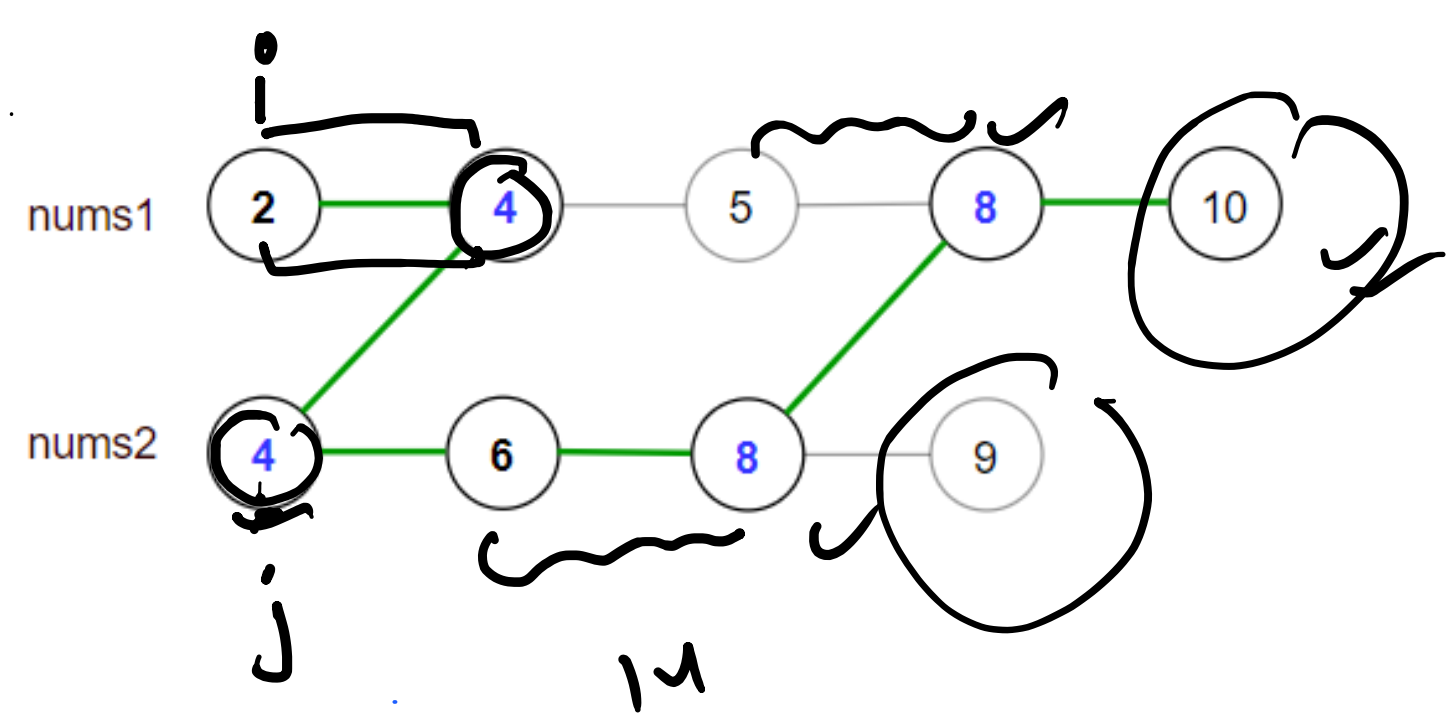
① Find PL
② Find RL
③ P 89
④ P 1 to n-1 Remove

next greater

2 3 4 6 7 2 6 6 3 2 2

① for i = n-2; i >= 0; i--
if (arr[i] < arr[i+1]) {
 z = i;
 break;
}

② for i = n-1; i >= 0; i--
if (arr[i] > arr[i+1]) {
 z = i;
 break;
}



6 + 14 + 10 = 30 ✓

13 + 18 + 22 + 64 = 122 ✓

2 3 7 10 12 15 30 34
1 5 7 8 10 15 16 19

ans = 3 + 18 + 27

```
public static int Maximum_Score(int[] arr1, int[] arr2) {  
    int i = 0, j = 0, s1 = 0, s2 = 0, ans = 0;  
    while (i < arr1.length && j < arr2.length) {  
        if (arr1[i] < arr2[j]) {  
            i++;  
        } else if (arr1[i] > arr2[j]) {  
            j++;  
        } else {  
            int sum1 = sumofarray(arr1, s1, i);  
            int sum2 = sumofarray(arr2, s2, j);  
            ans += Math.max(sum1, sum2);  
            s1 = i + 1;  
            s2 = j + 1;  
            i++;  
            j++;  
        }  
    }  
}
```

n → Rick
m → cab

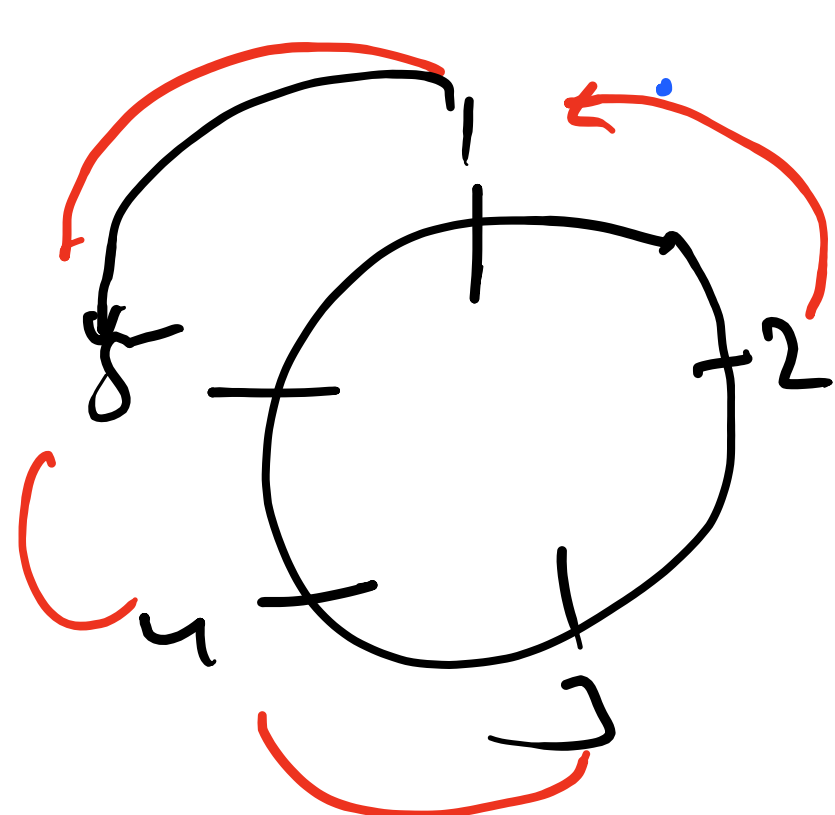
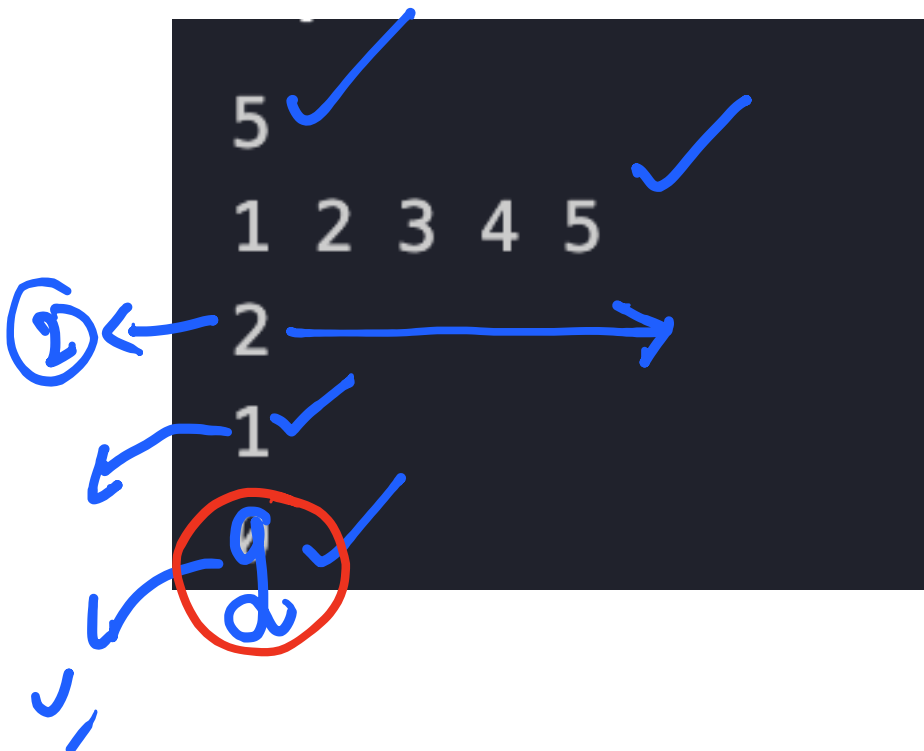
A C
C1 → 1 1
C2 → R C
C3 → A C ✓
C4 → R C ✓

2 + 5 = 7

n=2 m=3 c1 c2 c3 c4
R C 1 3 7 19
2 5 4 4 4

[2xc1, c2] = 2
[sx c1, c2] = 3/5
[4xc1, c2] = 3
[4xc1, c2] = 3/5
[5, c3] = 5
[9, c3] = 7

```
public static int minimumsum(int c1, int c2, int c3, int c4, int[] rickshaw_ride, int[] cab_ride) {  
    int rickshaw_cost = 0;  
    for (int i = 0; i < rickshaw_ride.length; i++) {  
        rickshaw_cost = rickshaw_cost + Math.min(rickshaw_ride[i] * c1, c2);  
    }  
}
```



x=2

1 2 3 4 5

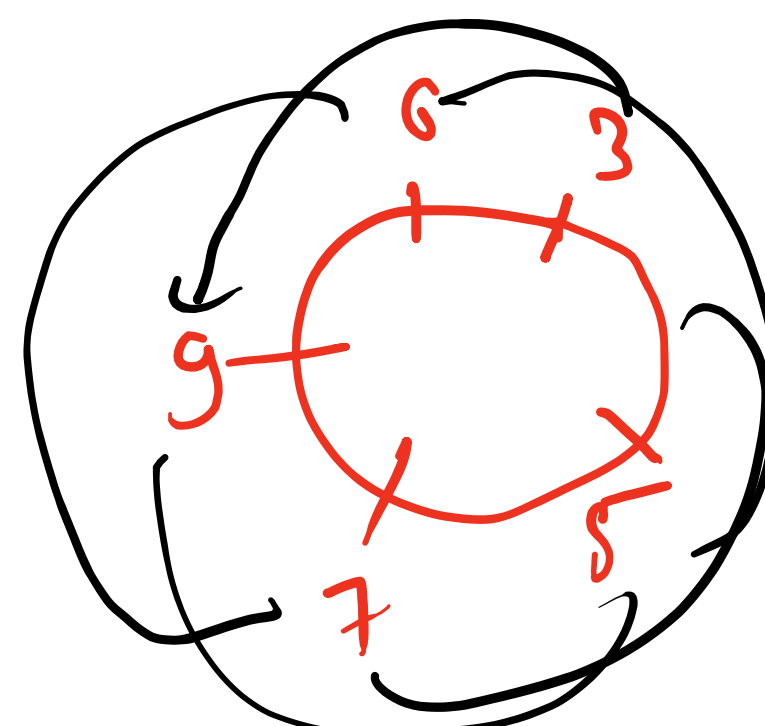
5 7 4 6 8

1 2 3 4 5

6 3 5 7 9

1 2 3 4 5

13 12 11 10 14



```
public static void main(String[] args) {  
    // TODO Auto-generated method stub  
    Scanner sc = new Scanner(System.in);  
    int n = sc.nextInt();  
    int[] arr = new int[n];  
    for (int i = 0; i < arr.length; i++) {  
        arr[i] = sc.nextInt();  
    }  
    int q = sc.nextInt();  
    while (q > 0) {  
        int x = sc.nextInt();  
        arr = modify(arr, x);  
        q--;  
    }  
}
```

arr

1 2 3 4 5

6 3 5 7 9

0 1 2 3 4

13 12 11 10 14

100%

X=2

i=0 0-2 = -2 + 5 = 3 ✓
i=1 1-2 = -1 + 5 = 4
i=2 2-2 = 0
i=3 3-2 = 1
i=4 4-2 = 2

6 7 8 9 4

14 10 12

```
public static int[] modify(int[] arr, int x) {  
    int[] a = new int[arr.length];  
    for (int i = 0; i < arr.length; i++) {  
        int j = i - x;  
        if (j < 0) {  
            a[i] = arr[i] + arr[j];  
        }  
    }  
    return a;  
}
```