Lec-27

06/04/24 10:11 AM

[50, 5, 3, 2, 8, 7, 9, 11, 3]

-1 8 8 8 9 9 11 -1 -1

$O(n^2) \rightarrow O(n)$

[50, 5, 3, 2, 8, 7, 9, 11, 3]

-1 8 8 8 9 9 11 -1 -1

arr[i]

for(i=0; i<arr.len; i++){

while(!st.isEmpty() && arr[i] > arr[st.peek()]){

~~ans[st.pop()] = arr[i]~~

}

st.push(i)

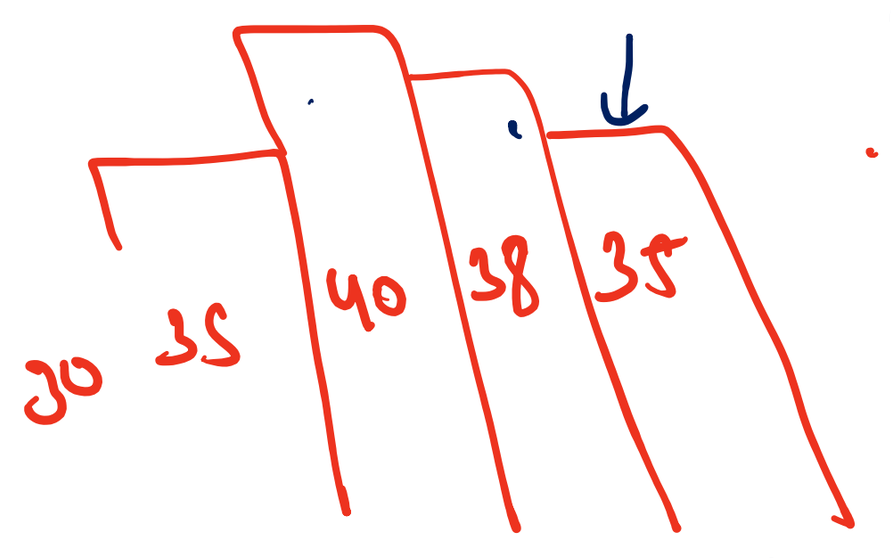
}

O(n)

dailyTemp

30 35 40 38 35

1 2 3 1 1



2 3 1 1

sum

[3, 1, 6, 4, 18, 7, 5, 11, 9]

for(i=0; i<arr.len; i++){

while(!st.isEmpty() && arr[i] > arr[st.peek()]){

st.pop()

}

span

if(st.isEmpty()){

ans[i] = i+1;

else{

ans[i] = i - st.peek()

}

st.push(i)

}

6-5=1

1-0=1

3-2=1

5-4=1

7-4=3

```java
public static void Cal_Span(int[] arr) {
        int[] ans= newint[arr.length];
        Stack<Integer> st= newStack<>();
        for(inti= 0; i< arr.length; i++) {
            while(!st.isEmpty() && arr[i] > arr[st.peek()]) {
                st.pop();
            }
            if(st.isEmpty()) {
                ans[i] = i+ 1;
            } else{
                ans[i] = i- st.peek();
            }
            st.push(i);
        }
        for(inti= 0; i< ans.length; i++) {
            System.out.print(ans[i] + " ");
        }
    }
```

[3, 1, 6, 4, 18, 7, 5, 11, 9]

1 1 3 5 1 1 3

[2, 3, 5, 4, 6, 1, 7]

2×5 = 10

3×4 = 12

5×1 = 5

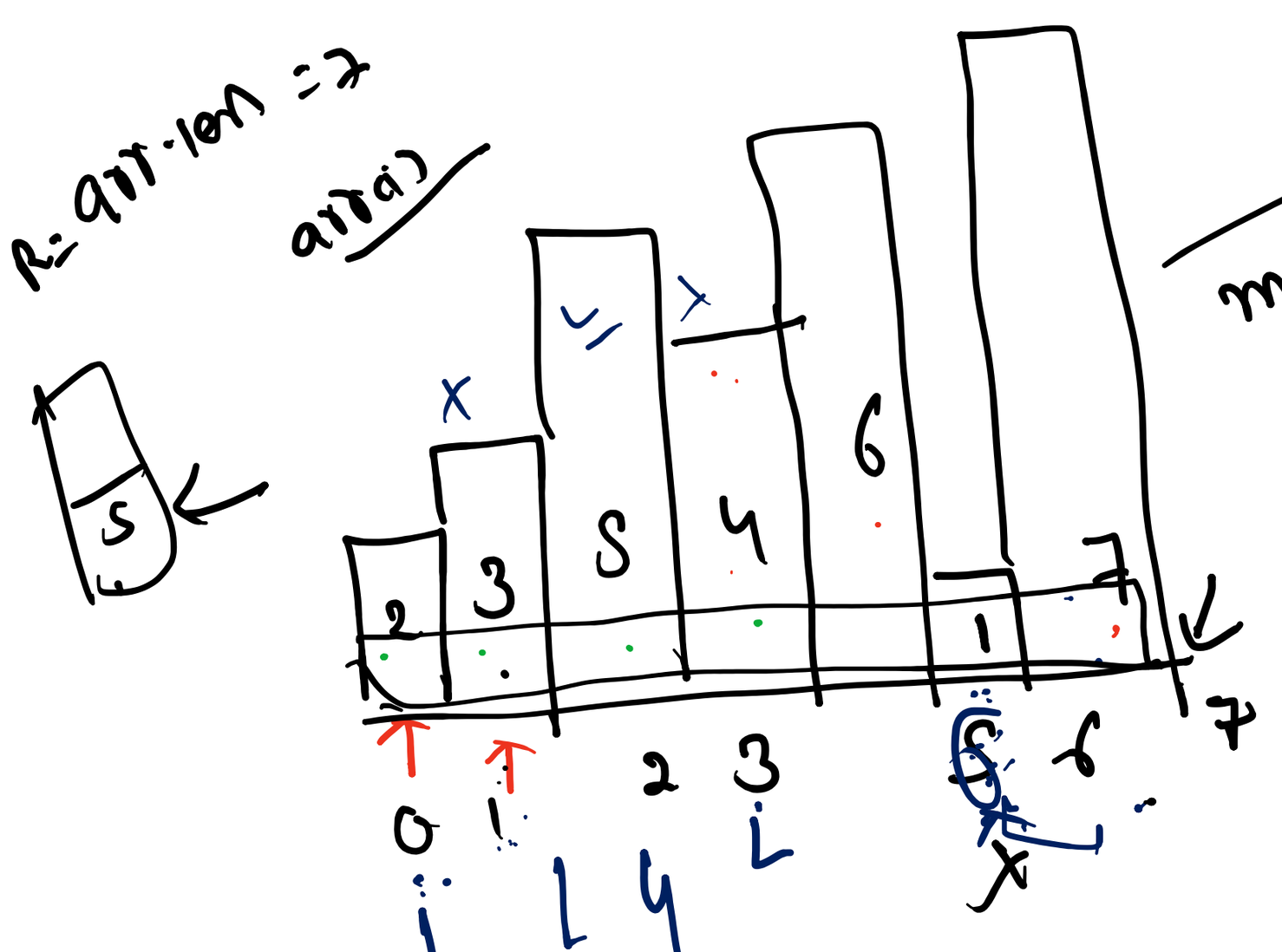
4×3 = 12

6×1 = 6

1×7 = 7

7×1 = 7

R = arr.len = 7



7×(7-5-1)=7

1×(7)

R=8

4×(5-1-1)=12

3×(5-0-1)=12

2×5=10

5×(8-1-1)=5

6×(5-3-1)=6

ans = 12

for(i=0; i<arr.len; i++){

while(!st.isEmpty() && arr[i] < arr[st.peek()]){

int h = arr[st.pop()]

R = i

l = st.peek()

ans = max(ans, h×(R-l-1))

ans = max(ans, h×R)

}

st.push(i)

}

R=arr.len

R=7

[2, 3, 5, 4, 6, 1, 7]

R = arr.len = 7

R=5

```java
public static int Area(int[] arr) {
        Stack<Integer> st= newStack<>();
        intans= 0;
        for(inti= 0; i< arr.length; i++) {
            while(!st.isEmpty() && arr[i] < arr[st.peek()]) {
                inth= arr[st.pop()];
                intr= i;
                if(!st.isEmpty()) {

                    intl= st.peek();
                    ans= Math.max(ans, h* (r- l- 1));
                } else{
                    ans= Math.max(ans, h* r);
                }
            }
            st.push(i);

        }

    }
```

6×(5-3-1)=6

4×(5-1-1)=12

3×(5-0-1)=12

2×5 = 10