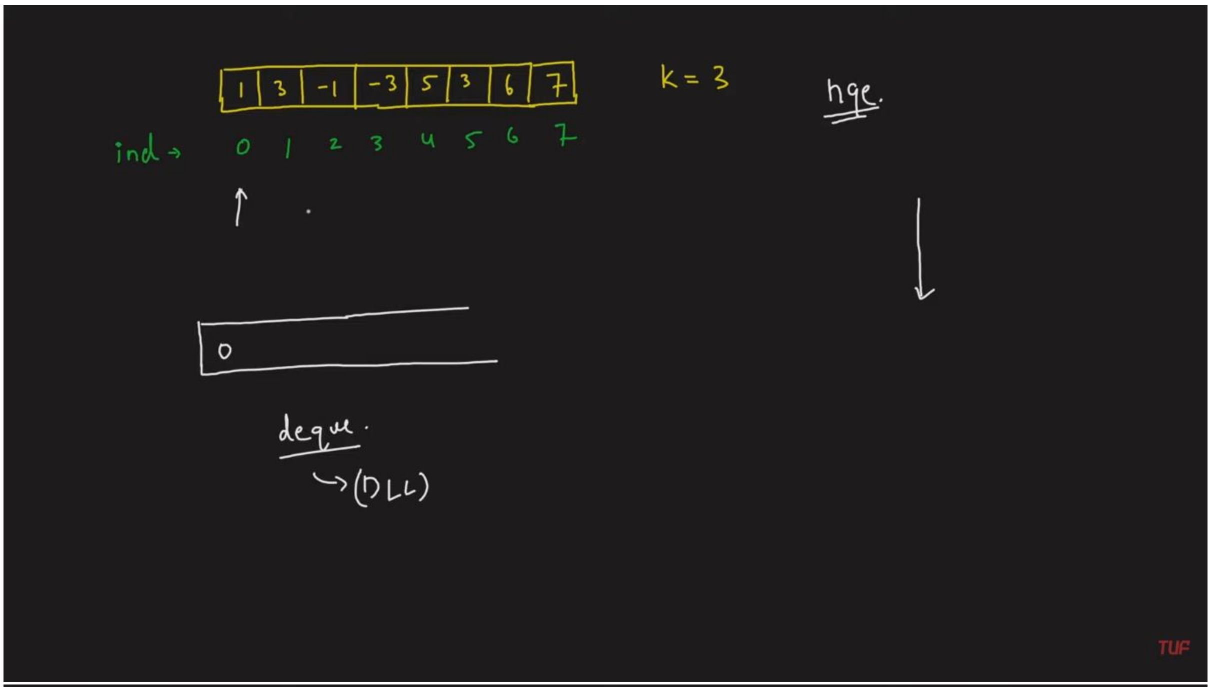
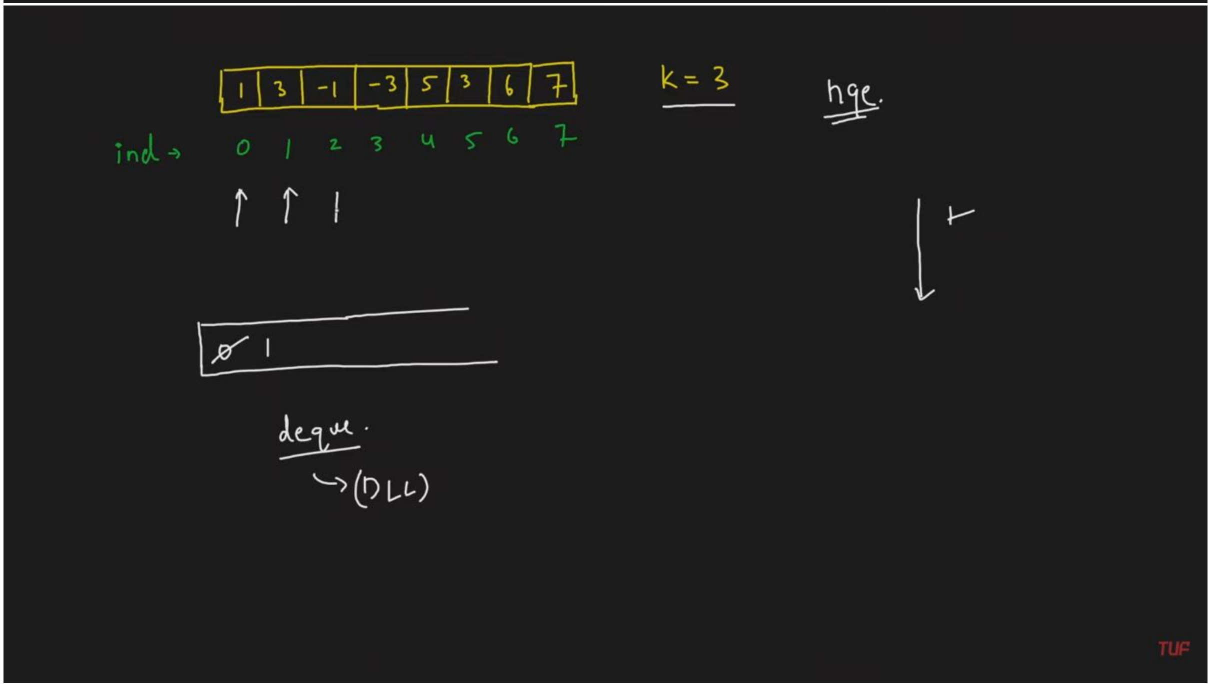
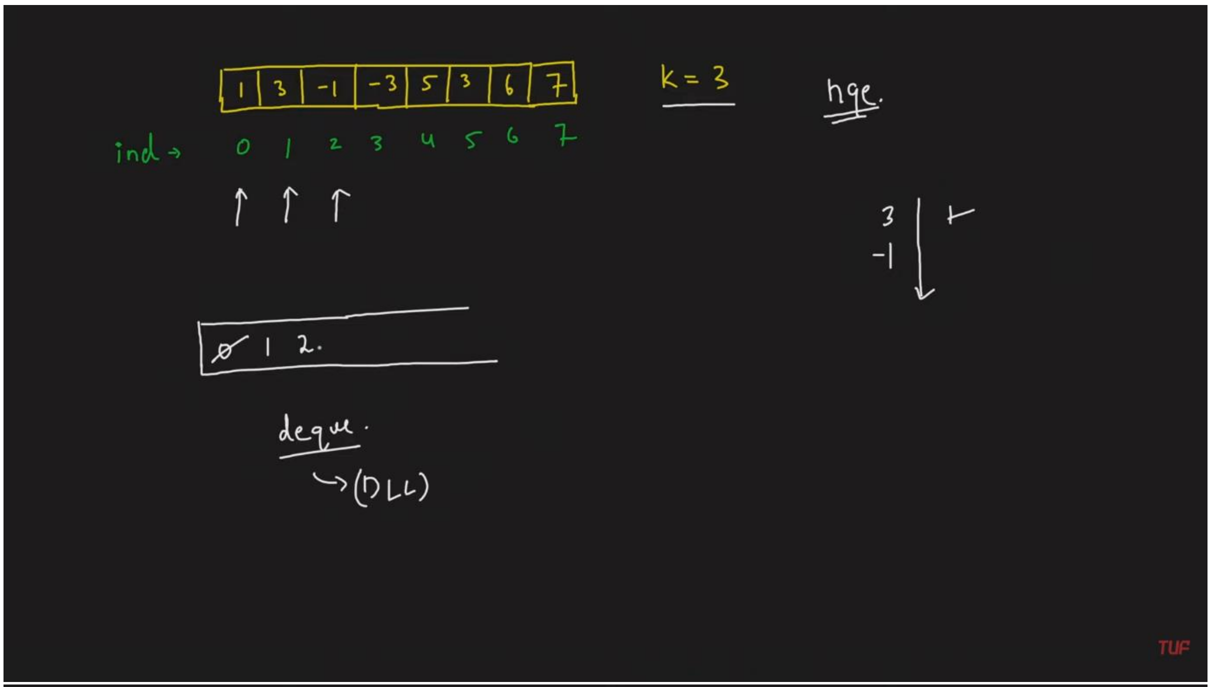
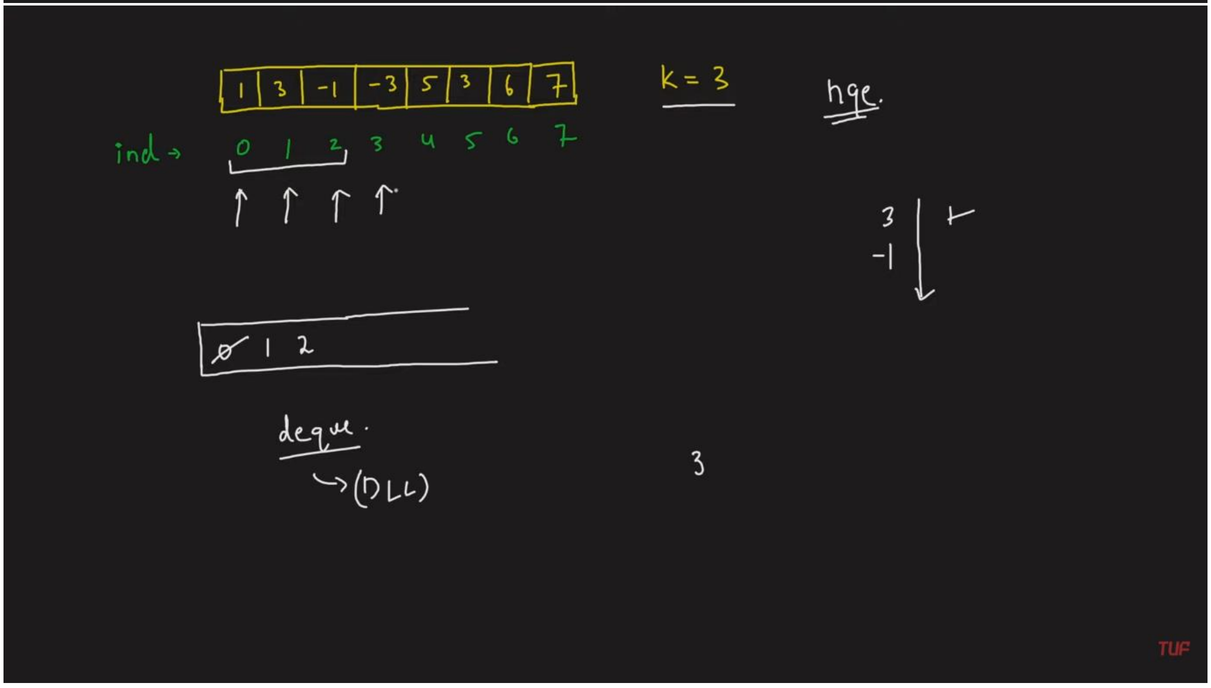

1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
ind → 0 1 2 3 4 5 6 7
k = 3
hge.
0
deque.
↪ (DLL)
TUF

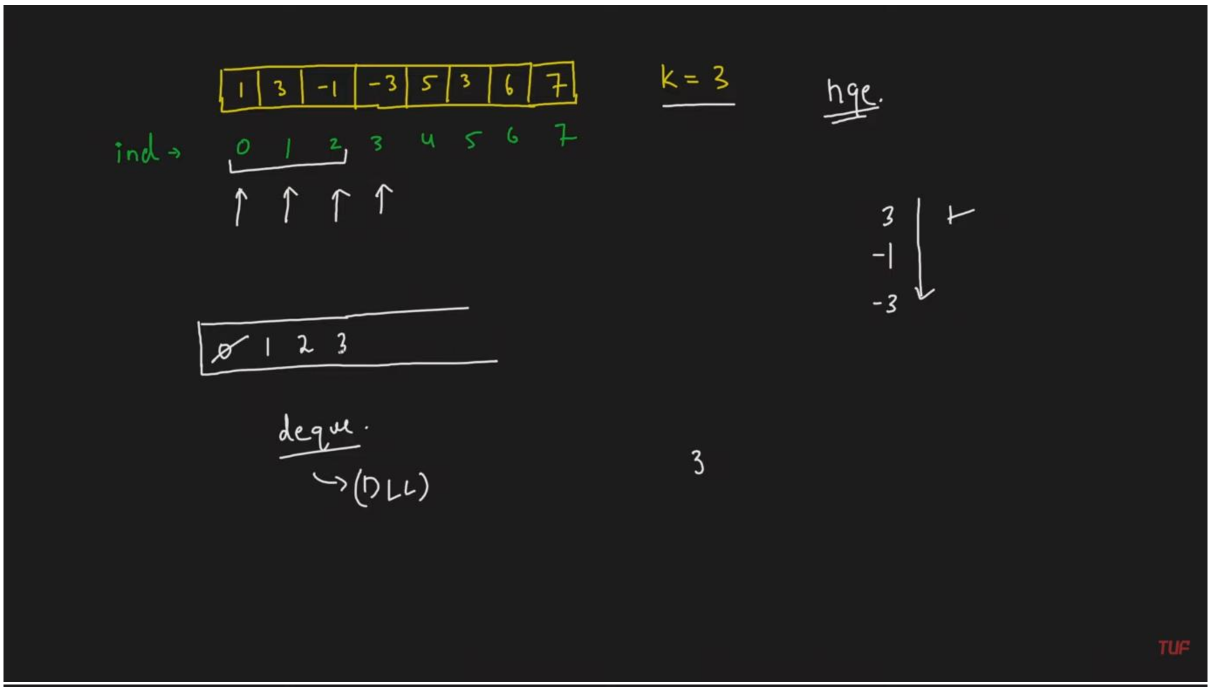

1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
ind → 0 1 2 3 4 5 6 7
k = 3
hge.
~~0~~ 1
deque.
↪ (DLL)
TUF

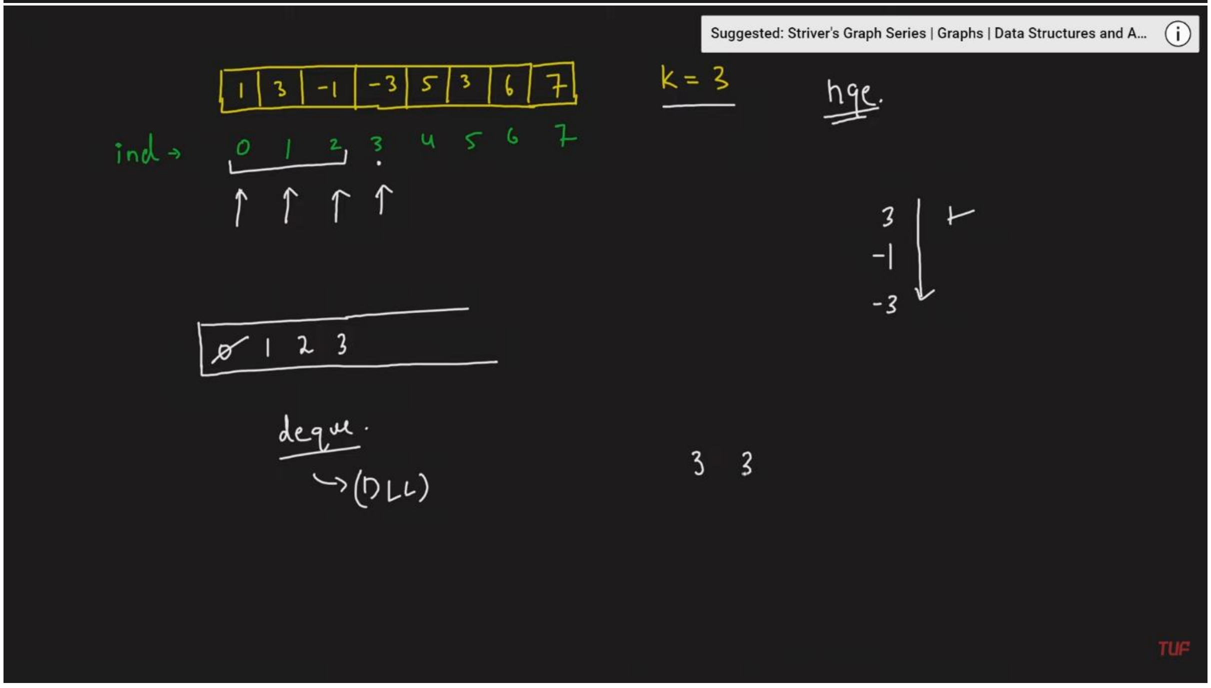


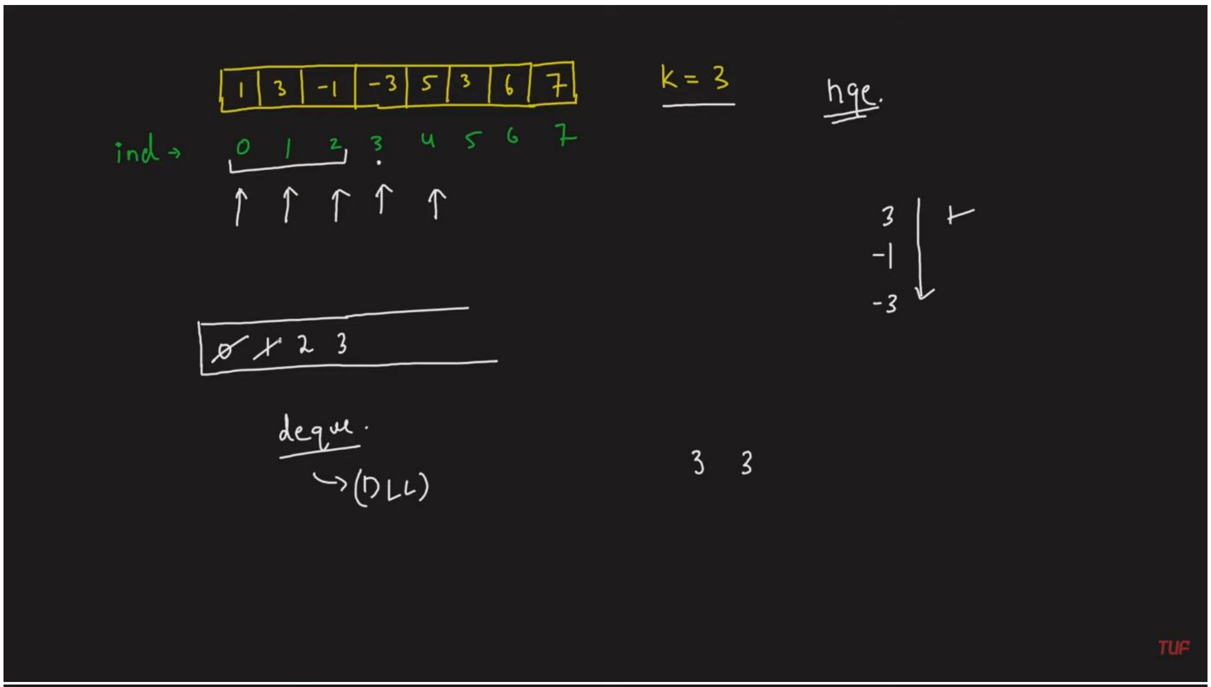
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
ind → 0 1 2 3 4 5 6 7
k = 3
hge.
3
-1
⊘ 1 2.
deque.
↪ (DLL)
TUF



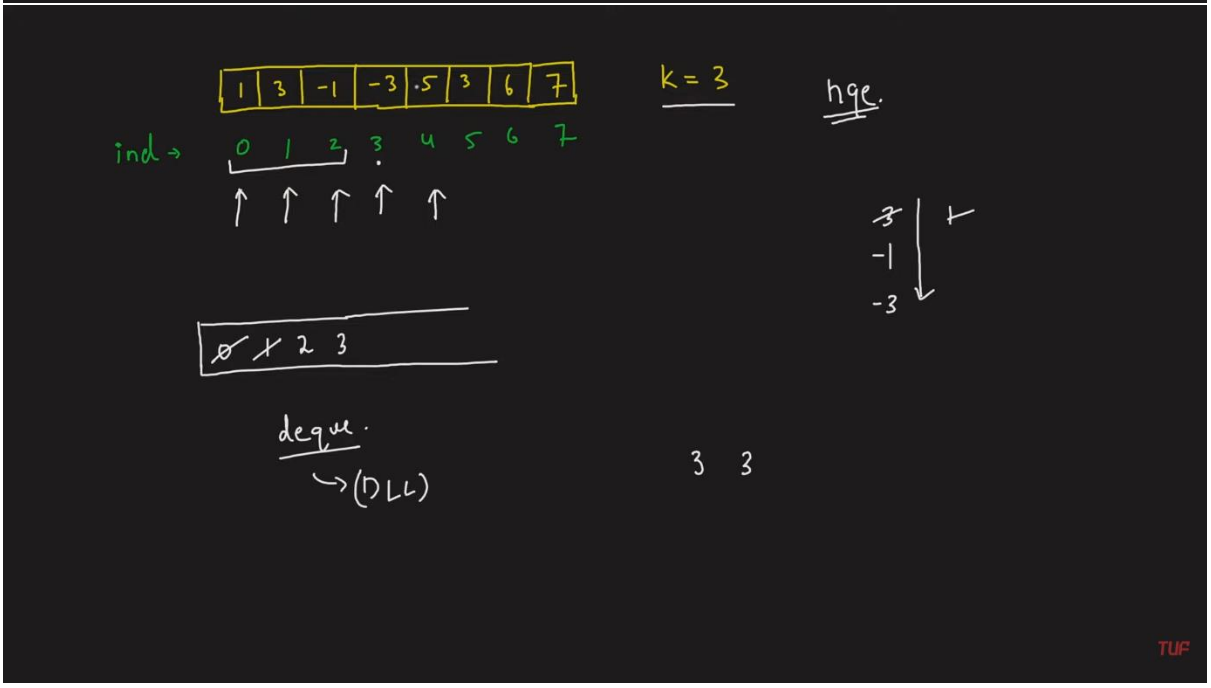
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
ind → 0 1 2 3 4 5 6 7
k = 3
hge.
3
-1
⊘ 1 2
deque.
↪ (DLL)
3
TUF



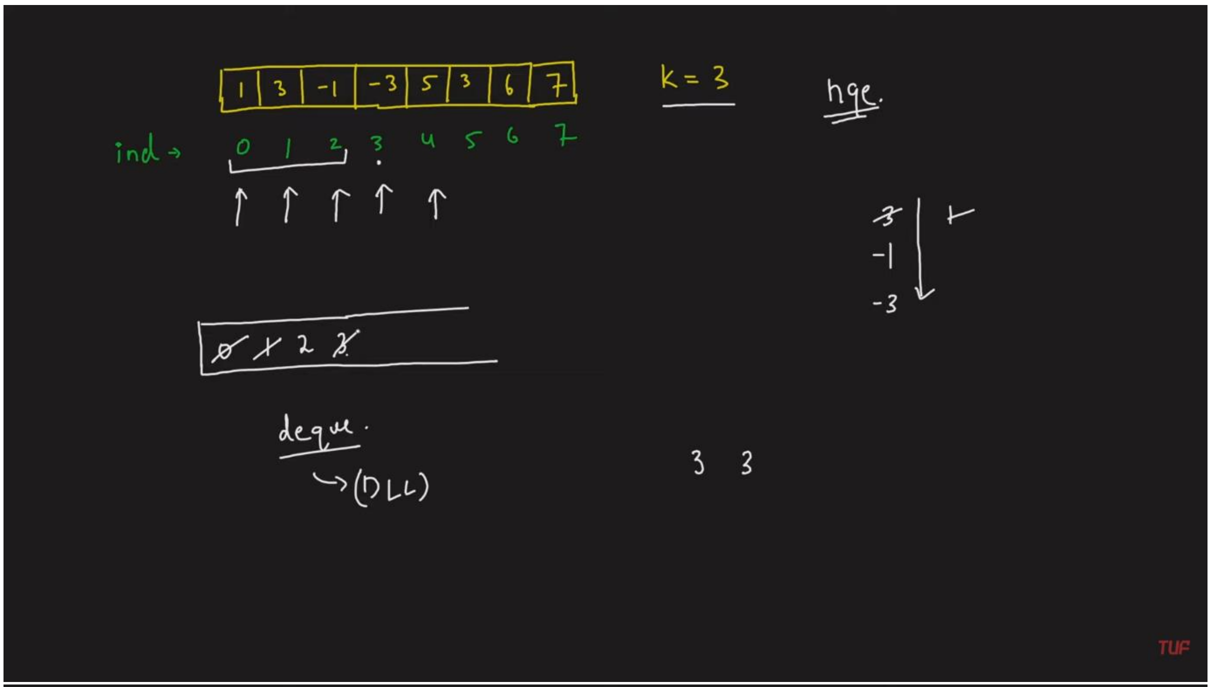

Suggested: Striver's Graph Series | Graphs | Data Structures and A...



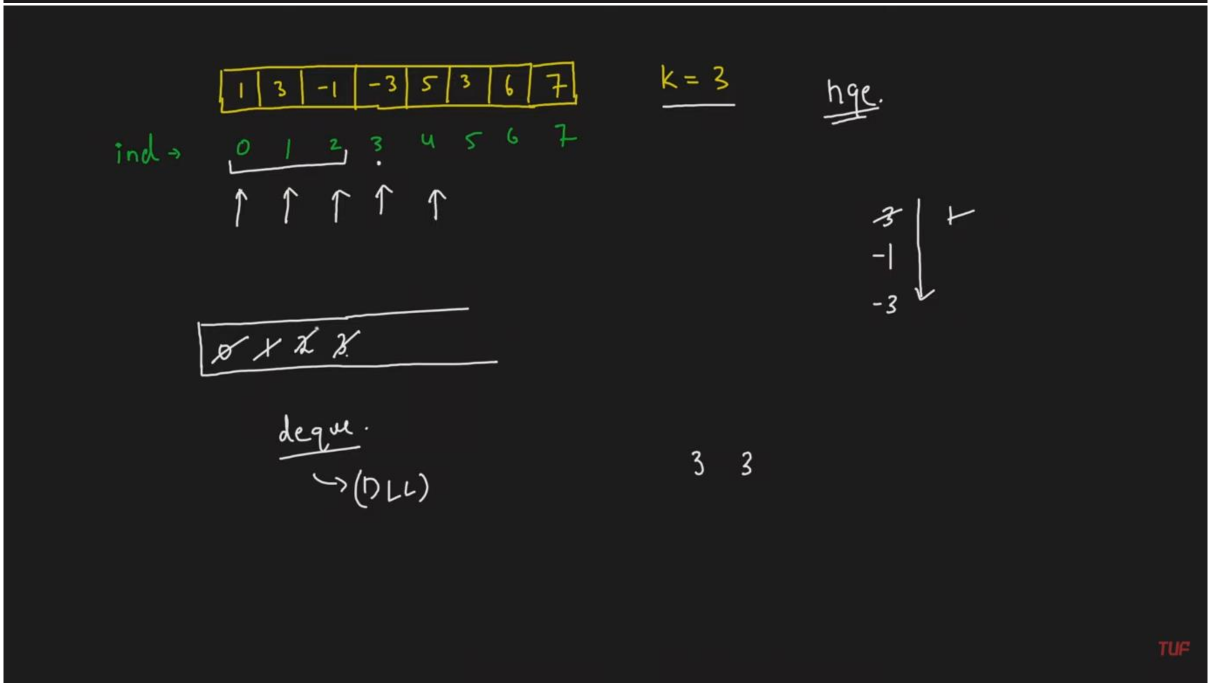
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
nge.
ind →
0 1 2 3 4 5 6 7
3
-1
-3
~~0~~ ~~1~~ 2 3
deque.
(DLL)
3 3
TUF



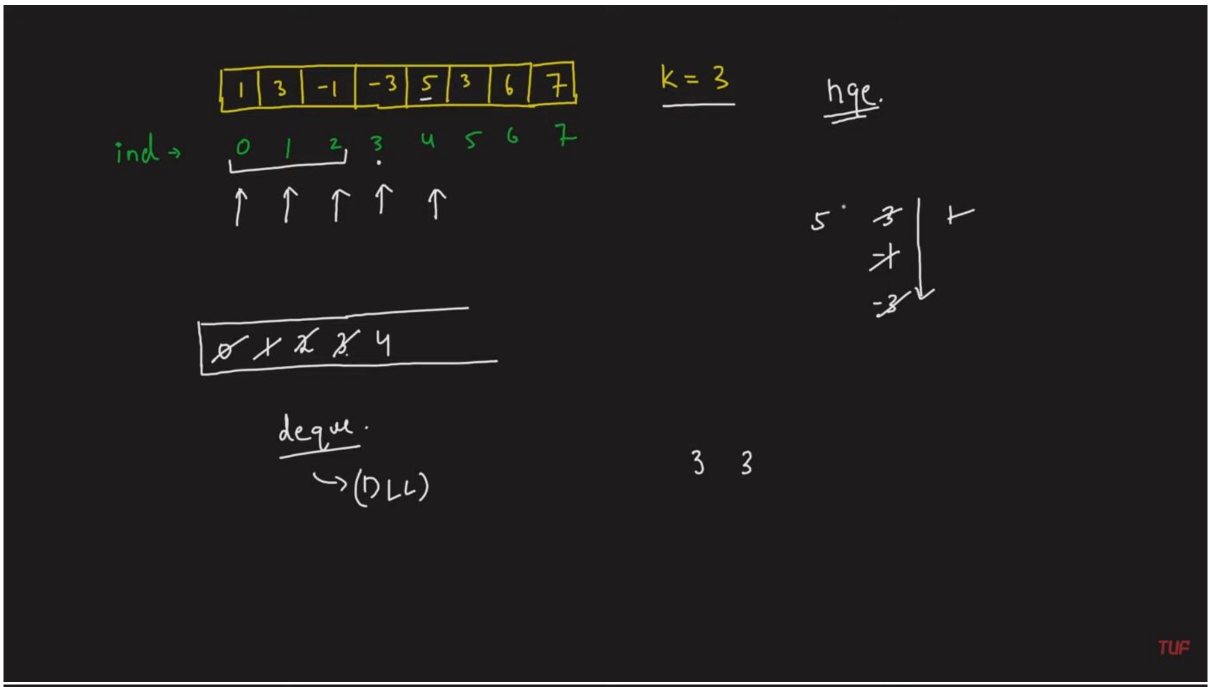
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
nge.
ind →
0 1 2 3 4 5 6 7
~~3~~
-1
-3
~~0~~ ~~1~~ 2 3
deque.
(DLL)
3 3
TUF

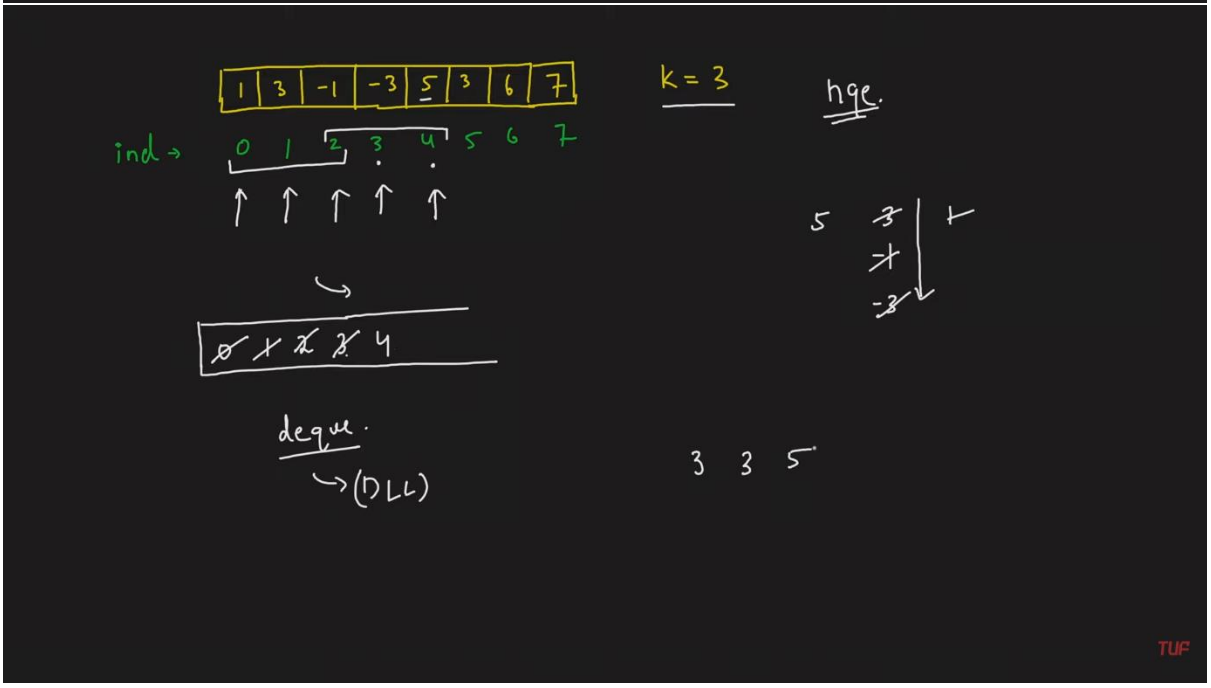


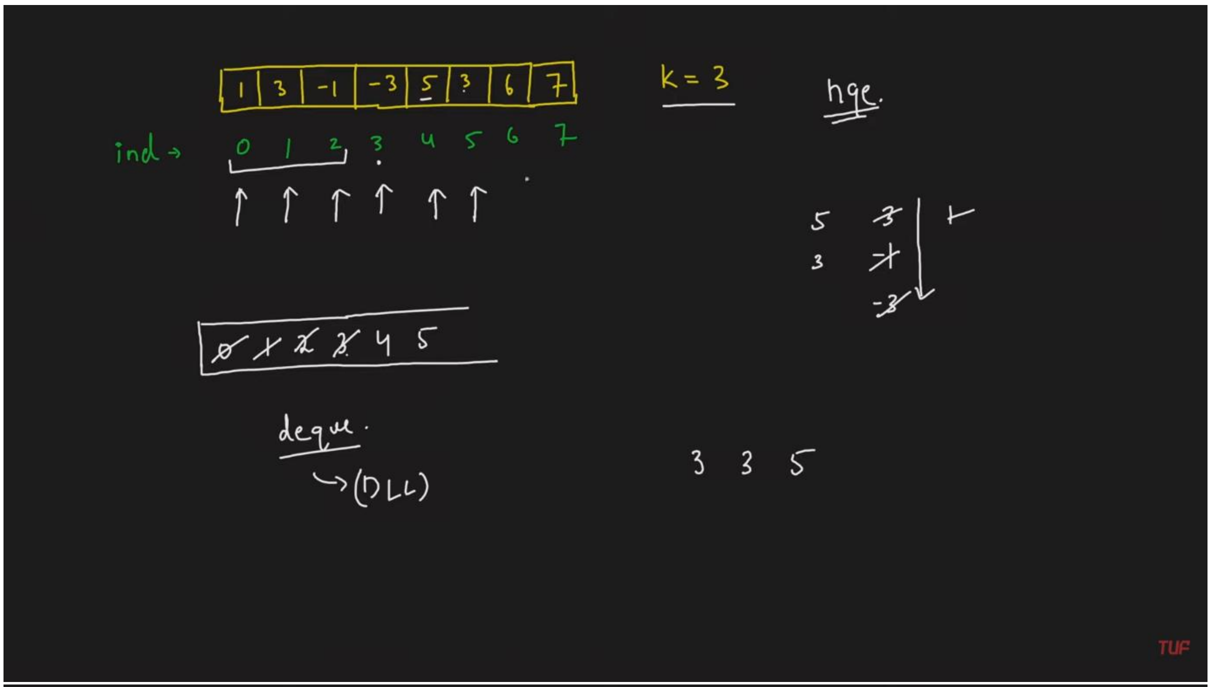
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
ind → 0 1 2 3 4 5 6 7
k = 3
nge.
~~3~~ -1 -3
~~0~~ ~~1~~ 2 ~~3~~
deque.
↪ (DLL)
3 3
TUF



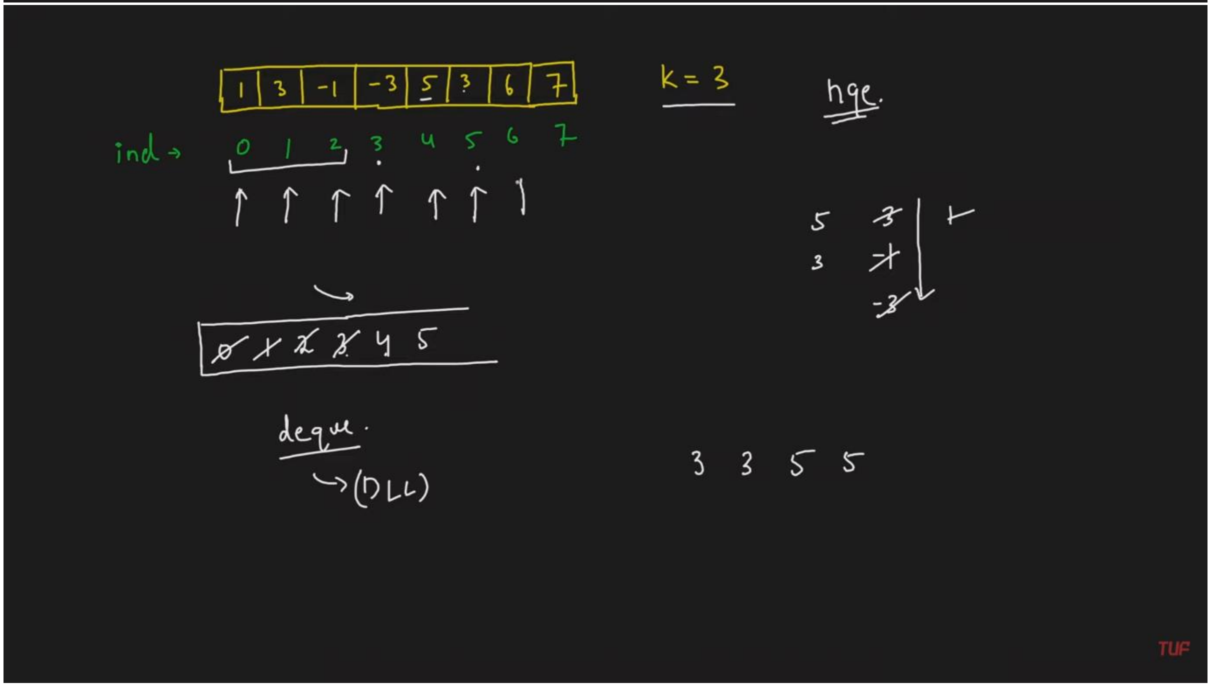
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
ind → 0 1 2 3 4 5 6 7
k = 3
nge.
~~3~~ -1 -3
~~0~~ ~~1~~ ~~2~~ ~~3~~
deque.
↪ (DLL)
3 3
TUF


1 3 -1 -3 5 3 6 7
ind → 0 1 2 3 4 5 6 7
k = 3
nge.
5 3 1 -3
0 1 2 3 4
deque.
↪ (DLL)
3 3

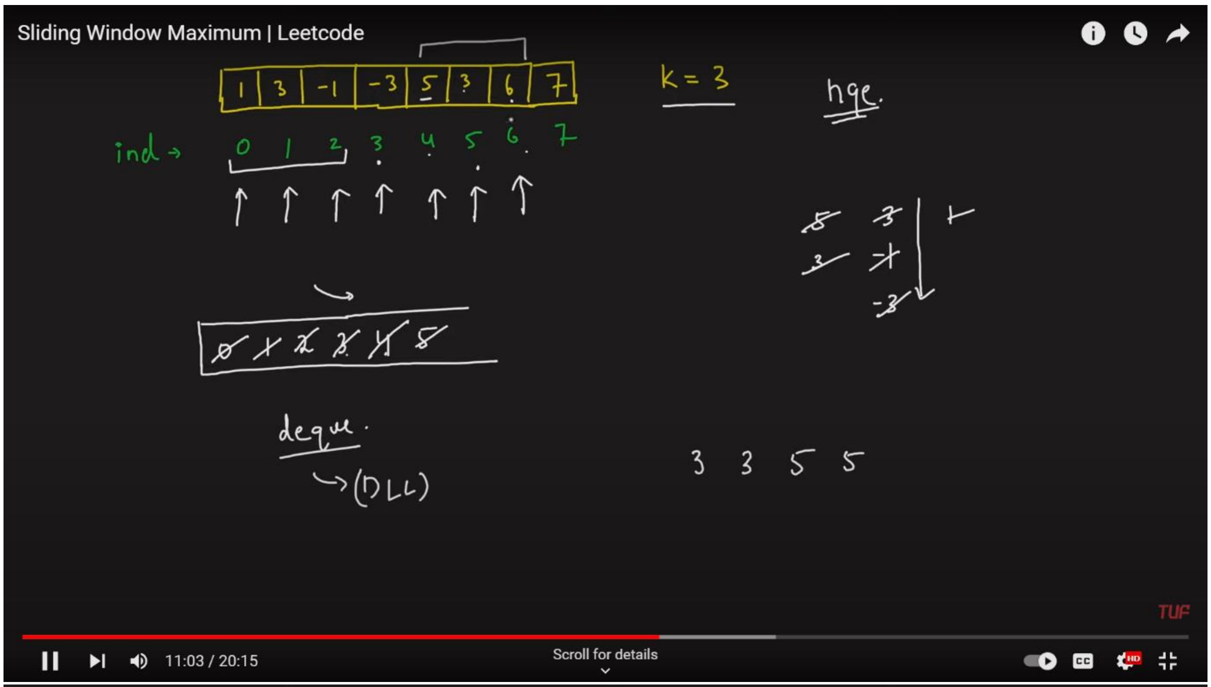

1 3 -1 -3 5 3 6 7
ind → 0 1 2 3 4 5 6 7
k = 3
nge.
5 3 1 -3
0 1 2 3 4
deque.
↪ (DLL)
3 3 5

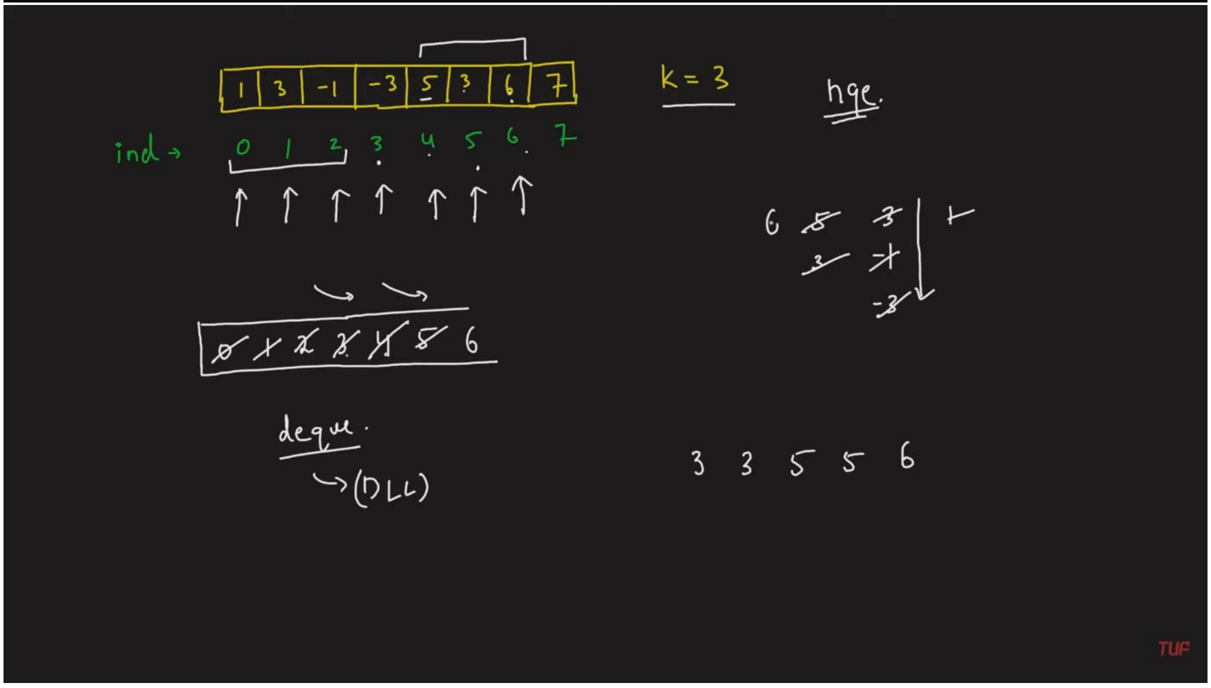


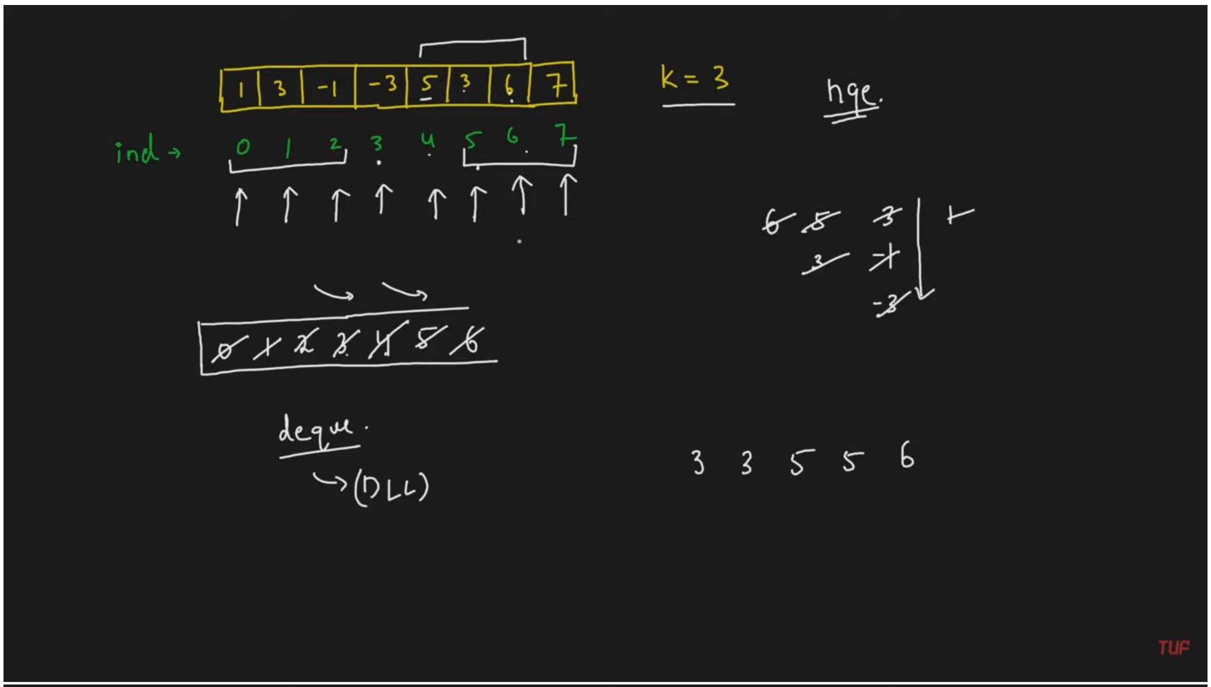
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
nge.
ind → 0 1 2 3 4 5 6 7
5 3
3 1
-3
0 1 2 3 4 5
deque.
→ (DLL)
3 3 5
TUF

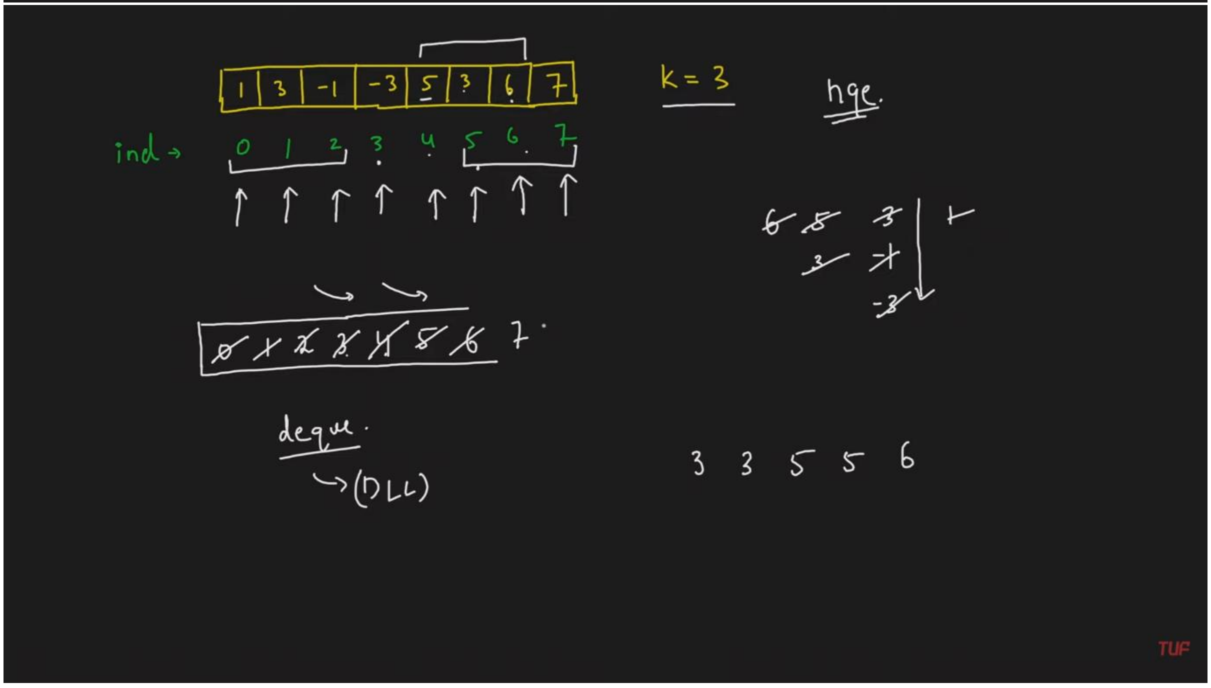


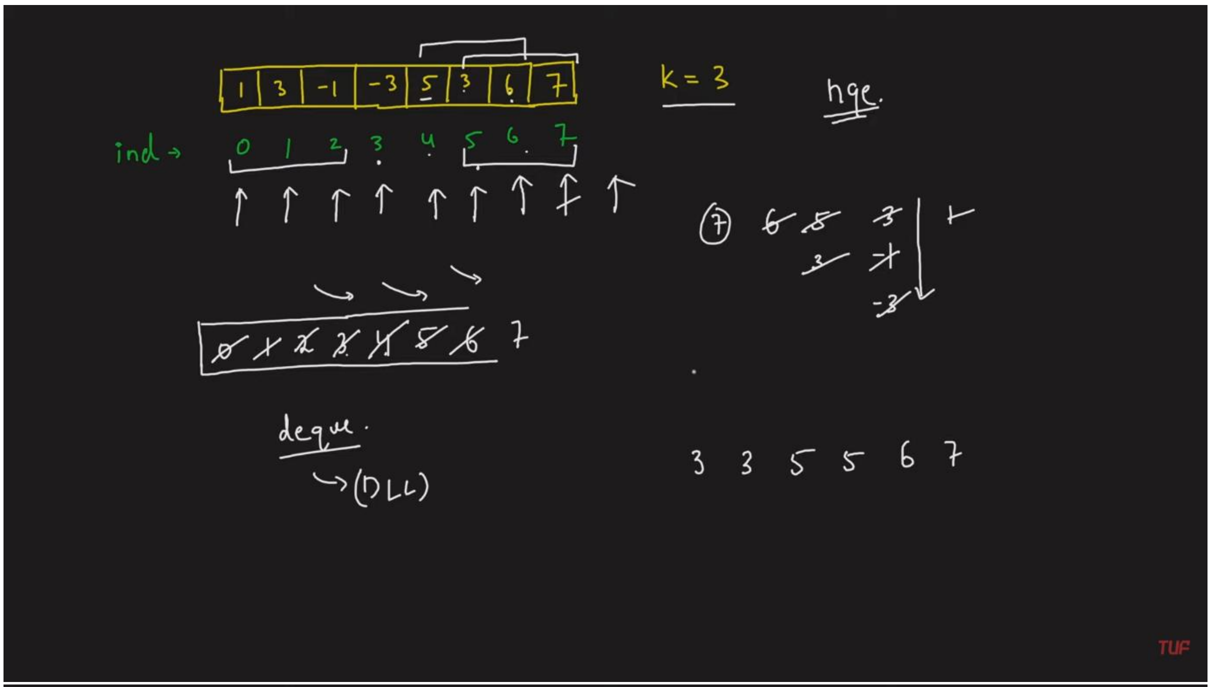
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
nge.
ind → 0 1 2 3 4 5 6 7
5 3
3 1
-3
0 1 2 3 4 5
deque.
→ (DLL)
3 3 5 5
TUF


Sliding Window Maximum | Leetcode
1 3 -1 -3 5 3 6 7
k = 3
nge.
ind → 0 1 2 3 4 5 6 7
deque.
↪ (DLL)
3 3 5 5
11:03 / 20:15
Scroll for details
TUF

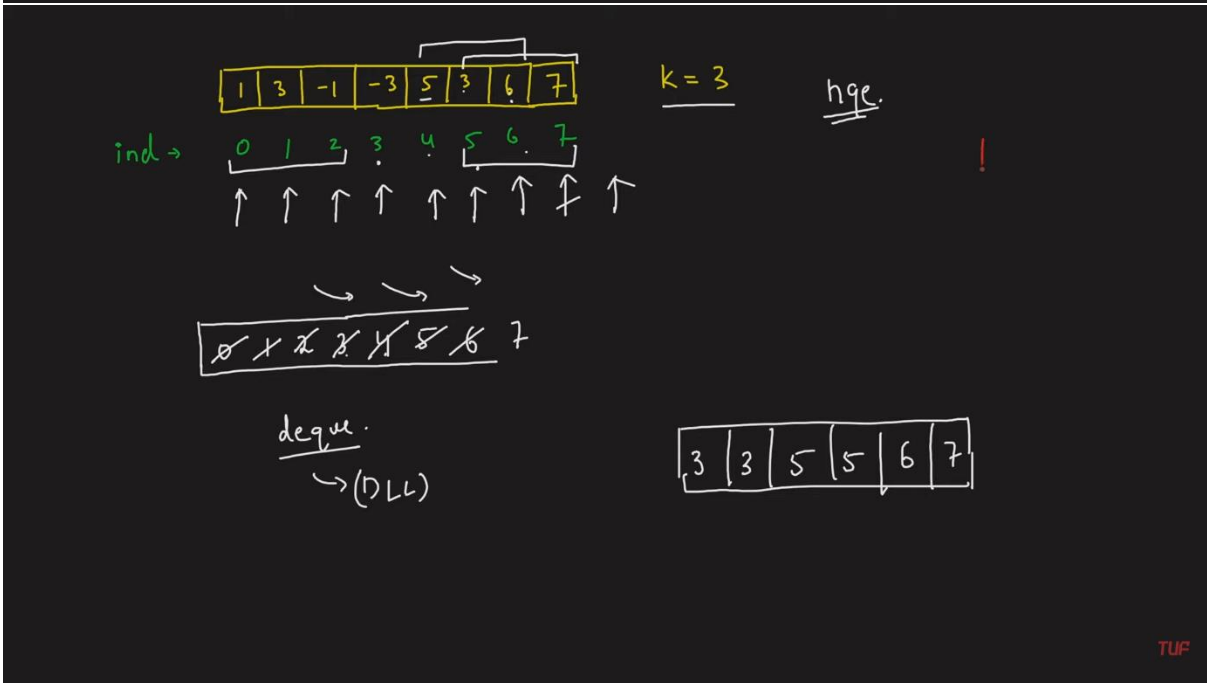

1 3 -1 -3 5 3 6 7
k = 3
nge.
ind → 0 1 2 3 4 5 6 7
6
deque.
↪ (DLL)
3 3 5 5 6
TUF


1 3 -1 -3 5 3 6 7
ind → 0 1 2 3 4 5 6 7
k = 3
nge.
deque.
(DLL)
3 3 5 5 6
TUF

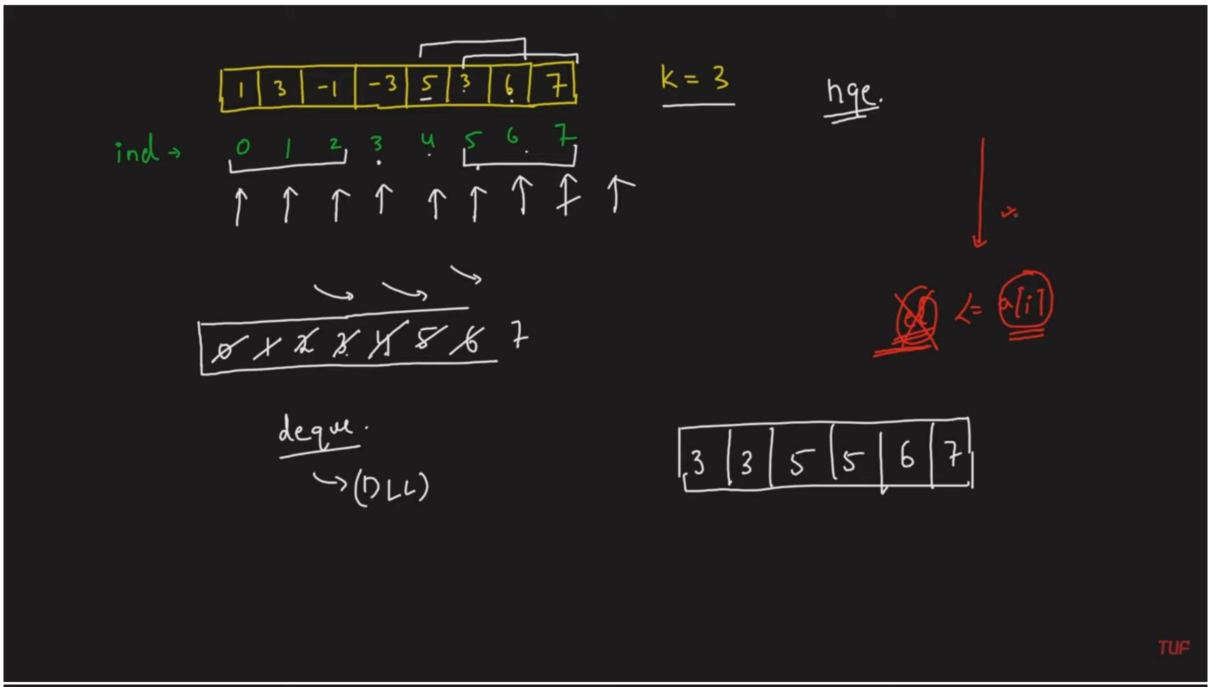

1 3 -1 -3 5 3 6 7
ind → 0 1 2 3 4 5 6 7
k = 3
nge.
7
deque.
(DLL)
3 3 5 5 6
TUF

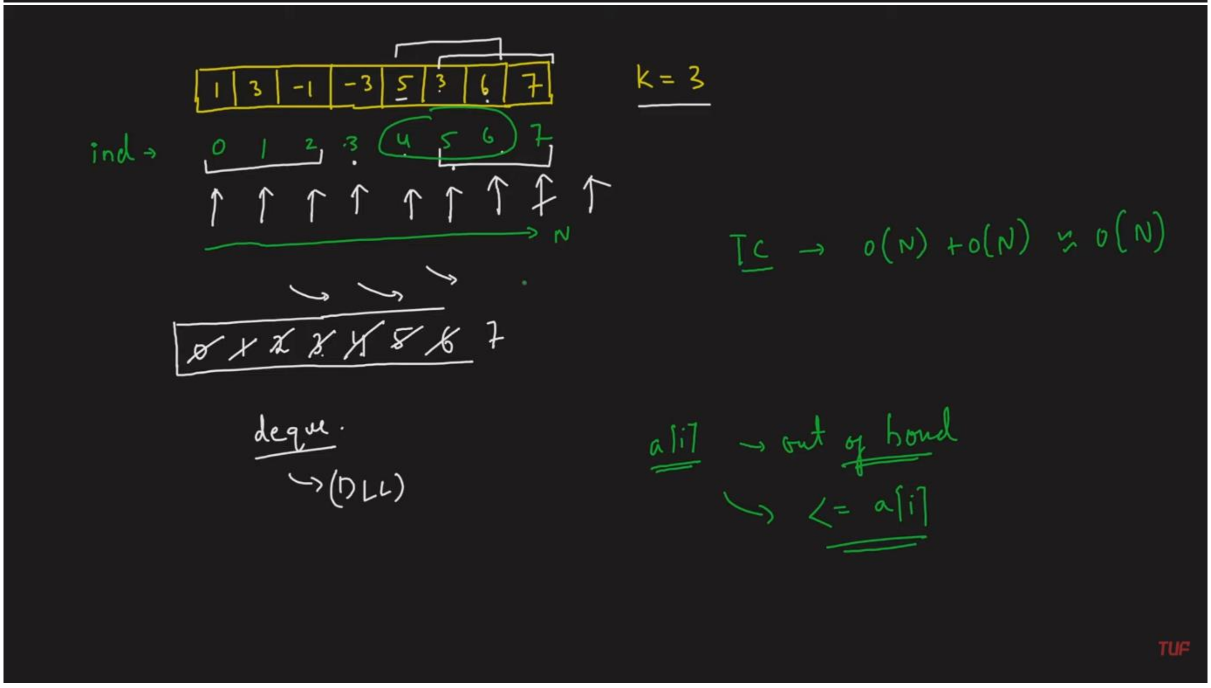


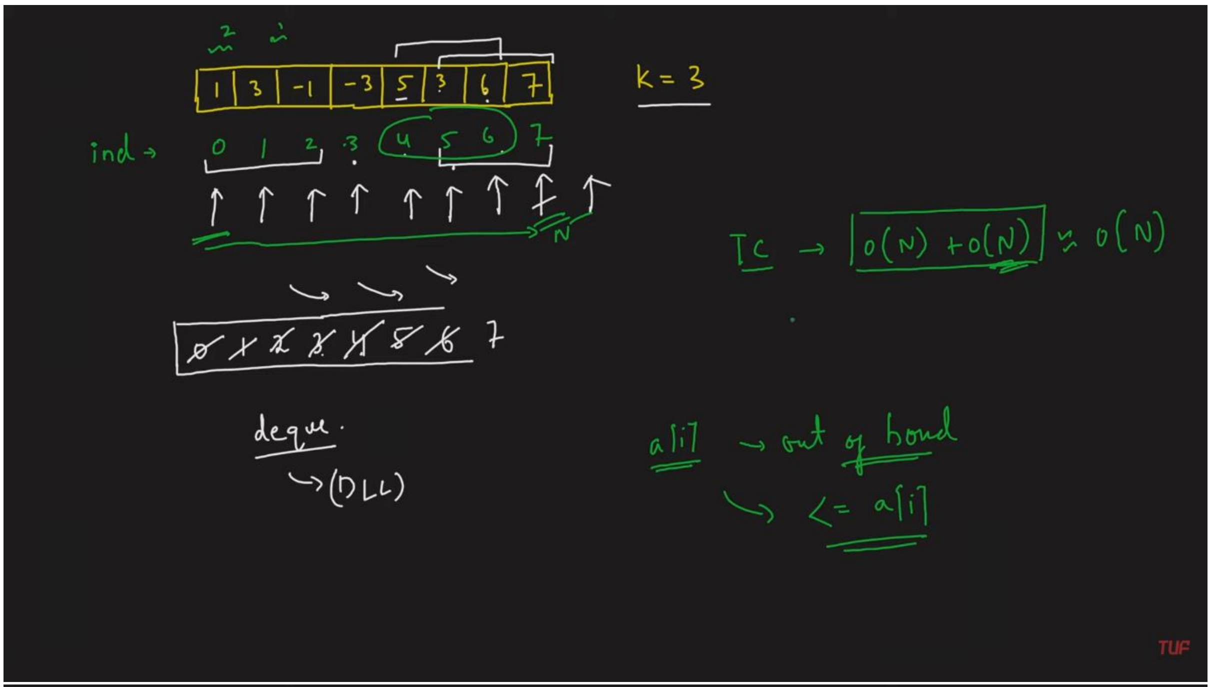
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
nge.
ind →
0 1 2 3 4 5 6 7
7 6 5 3
3 -1
-3
0 1 2 3 4 5 6 7
deque.
↪ (DLL)
3 3 5 5 6 7
TUF

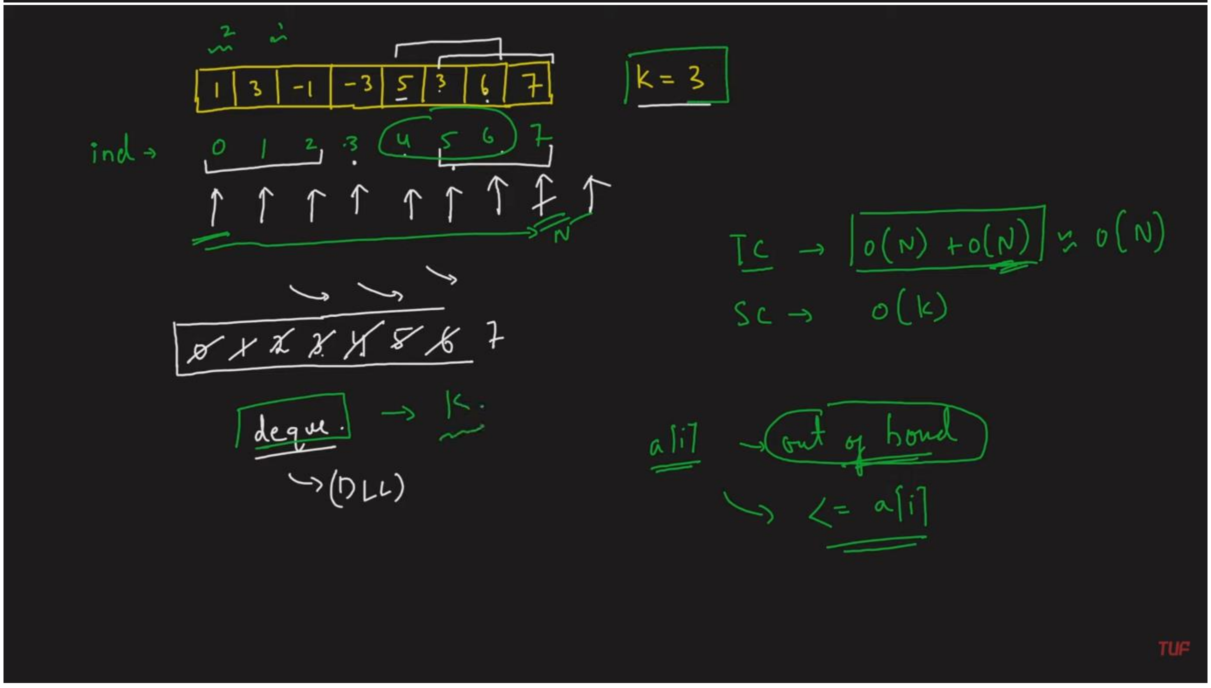


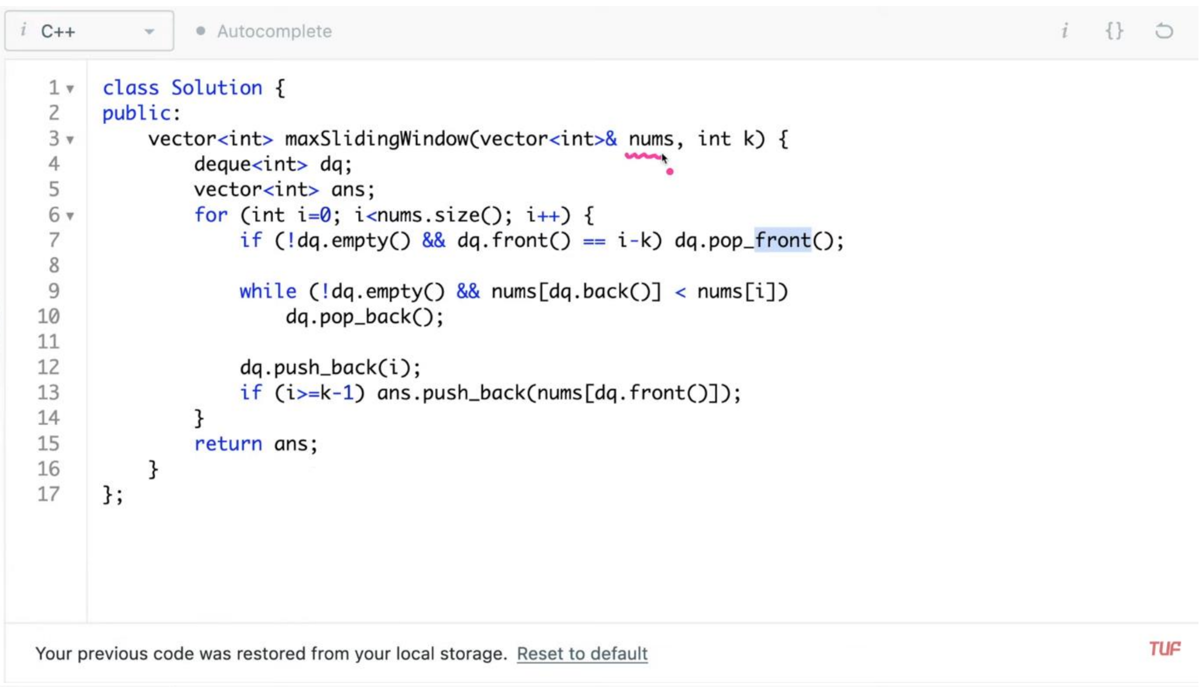
1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
nge.
ind →
0 1 2 3 4 5 6 7
!
0 1 2 3 4 5 6 7
deque.
↪ (DLL)
3 | 3 | 5 | 5 | 6 | 7
TUF


1 3 -1 -3 5 3 6 7
k = 3
nge.
ind → 0 1 2 3 4 5 6 7
0 1 2 3 4 5 6 7
$\cancel{a} <= a[i]$
deque.
→ (DLL)
3 3 5 5 6 7
TUF


1 3 -1 -3 5 3 6 7
k = 3
ind → 0 1 2 3 4 5 6 7
N
TC → O(N) + O(N) ≈ O(N)
0 1 2 3 4 5 6 7
deque.
→ (DLL)
a[i] → out of bound
→ <= a[i]
TUF


1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
ind → 0 1 2 3 4 5 6 7
N
0 1 2 3 4 5 6 7
deque.
↪ (DLL)
TC → O(N) + O(N) ≈ O(N)
a[i] → out of bond
↪ <= a[i]
TUF


1 | 3 | -1 | -3 | 5 | 3 | 6 | 7
k = 3
ind → 0 1 2 3 4 5 6 7
N
0 1 2 3 4 5 6 7
deque. → k.
↪ (DLL)
TC → O(N) + O(N) ≈ O(N)
SC → O(k)
a[i] → out of bond
↪ <= a[i]
TUF

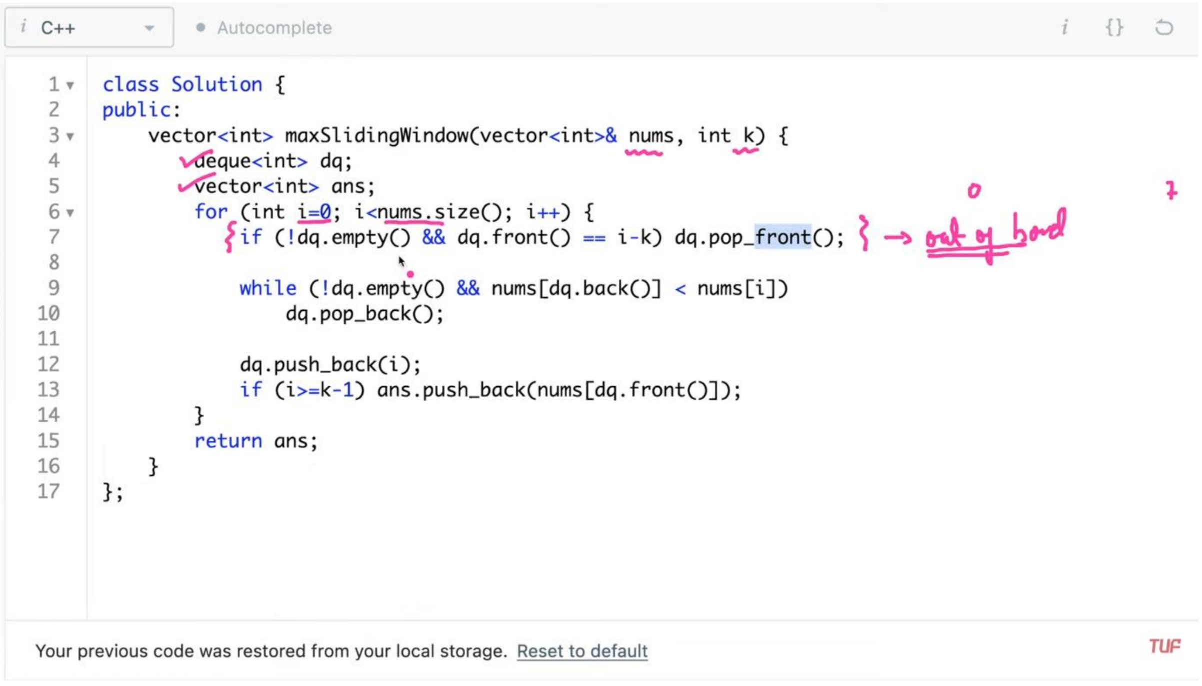


```cpp
class Solution {
public:
    vector<int> maxSlidingWindow(vector<int>& nums, int k) {
        deque<int> dq;
        vector<int> ans;
        for (int i=0; i<nums.size(); i++) {
            if (!dq.empty() && dq.front() == i-k) dq.pop_front();

            while (!dq.empty() && nums[dq.back()] < nums[i])
                dq.pop_back();

            dq.push_back(i);
            if (i>=k-1) ans.push_back(nums[dq.front()]);
        }
        return ans;
    }
};
```

Your previous code was restored from your local storage. Reset to default

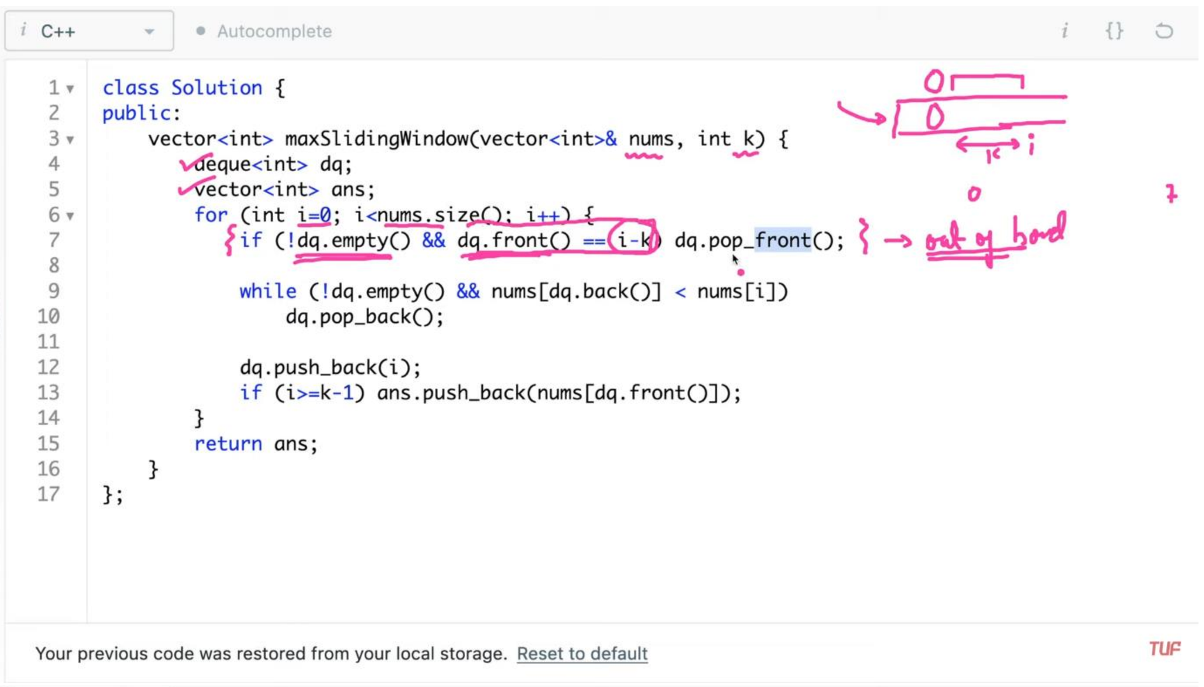


```cpp
class Solution {
public:
    vector<int> maxSlidingWindow(vector<int>& nums, int k) {
        deque<int> dq;
        vector<int> ans;
        for (int i=0; i<nums.size(); i++) {
            if (!dq.empty() && dq.front() == i-k) dq.pop_front();

            while (!dq.empty() && nums[dq.back()] < nums[i])
                dq.pop_back();

            dq.push_back(i);
            if (i>=k-1) ans.push_back(nums[dq.front()]);
        }
        return ans;
    }
};
```

Your previous code was restored from your local storage. Reset to default



C++ · Autocomplete

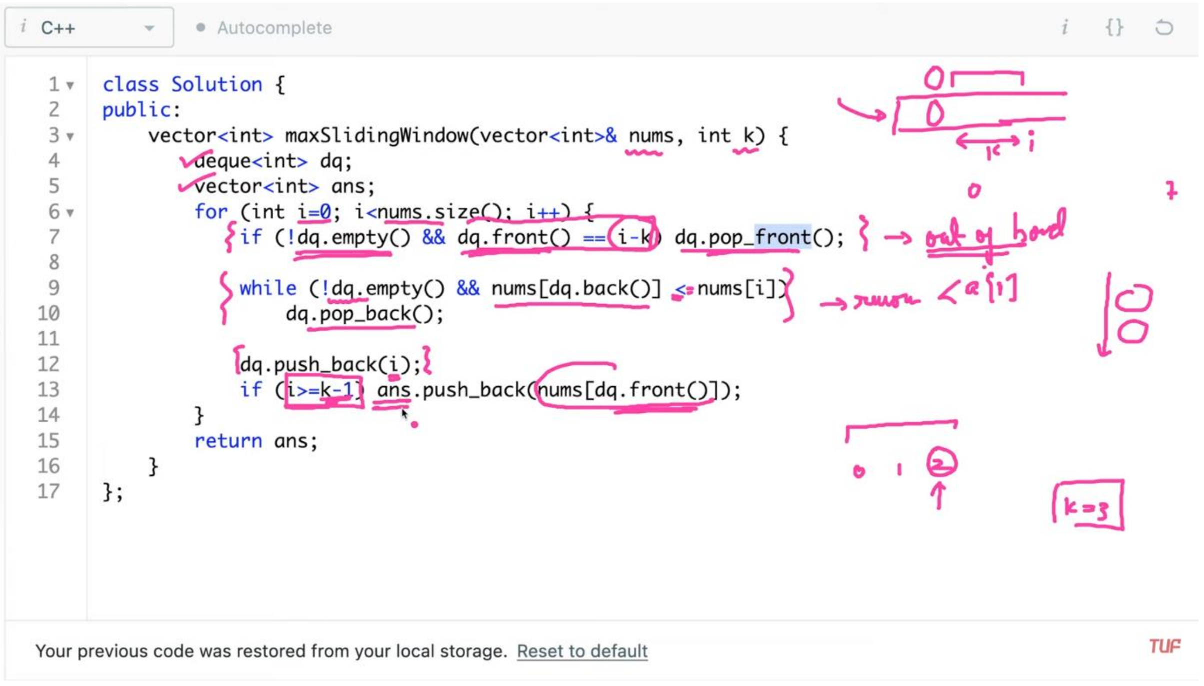
```cpp
class Solution {
public:
    vector<int> maxSlidingWindow(vector<int>& nums, int k) {
        deque<int> dq;
        vector<int> ans;
        for (int i=0; i<nums.size(); i++) {
            if (!dq.empty() && dq.front() == i-k) dq.pop_front();

            while (!dq.empty() && nums[dq.back()] < nums[i])
                dq.pop_back();

            dq.push_back(i);
            if (i>=k-1) ans.push_back(nums[dq.front()]);
        }
        return ans;
    }
};
```

Your previous code was restored from your local storage. Reset to default

TUF



C++ · Autocomplete

```cpp
class Solution {
public:
    vector<int> maxSlidingWindow(vector<int>& nums, int k) {
        deque<int> dq;
        vector<int> ans;
        for (int i=0; i<nums.size(); i++) {
            if (!dq.empty() && dq.front() == i-k) dq.pop_front();

            while (!dq.empty() && nums[dq.back()] < nums[i])
                dq.pop_back();

            dq.push_back(i);
            if (i>=k-1) ans.push_back(nums[dq.front()]);
        }
        return ans;
    }
};
```

Your previous code was restored from your local storage. Reset to default

TUF