

Sliding window

[1, 3, -1, -3, 5, 3, 6, 7]

Pair queue

vector ans

for (i=0; i < nums.size(); i++)

pq.push({nums[i], i})

if (i >= (k-1))

while (pq.top().second >= k)

pq.pop();

ans.push\_back(pq.top().first);

return ans

i=0

{1, 0}

i=2

ans [3, 3]

i=0 {1, 0} if x

i=1 {1, 0}, {3, 1} if x

i=2 {1, 0}, {3, 1}, {-1, 2}

if (2 >= (3-1))

while (2 - 2 >= k)

pq.pop()

i=2 {1, 0}, {3, 1} back

ans.push(3)

~~i=3 {1, 0}, {3, 1}, {-1, 3}~~

~~if (3 >= (2))~~

~~pop() -> {-3, 3}~~

~~ans.push(3)~~

~~i=4 {1, 0}, {3, 1}, {5, 4}~~

~~i=4 {1, 0}, {3, 1}, {5, 4}~~

~~if (4 >= 2)~~

~~while(4~~



$i=2 \{(3,1), (1,0), (-1,2)\}$

max heap logaya  
tha to

ans [ ]

[1, 3, -1, -3, 5, 3, 6, 7]

if ( $i \geq k-1$ ) ✓

while ( $i - pq.top().second() \geq k$ )

$(2 - 0 \geq 3) \times =$

ans = [3]

$i=3 \{(3,1), (1,0), (-1,2), (-3,3)\}$

ans = [3]

if ( $i \geq 2$ )

while ( $3 - 0 \geq 3$ ) ✓  $\{(1,0), (3,1), (-1,2), (-3,3)\}$

pq.pop();

ans.push(3)

ans = [3, 3]

pq.top

3, 1
-1, 2
-3, 3

$i=4 \{(3,1), (-1,2), (-3,3), (5,4)\}$

if ( $i \geq 2$ ) ✓

while ( $4 - 1 \geq 3$ ) ✓

~~pop~~ pq.pop()

[3, 1] remove out

ans.push(5);

pq.top

ans = [3, 3, 5]

5, 4
✓
✓

$i=5$

$\{(-1,2), (-3,3), (5,4), (3,5)\}$

if ( $i \geq 2$ ) ✓

while ( $5 - 2 \geq 3$ ) ✓

pq.pop

$(-1,2)$  remove out

5, 4

ans.pop(5)

ans = [3, 3, 5, 5]

$i=6 \{(3,3), (5,4), (3,5), (6,6)\}$

if ( $i \geq 2$ )

while ( $6 - 3 \geq 3$ )

pq.pop( $(-3,3)$  remove out)

6, 6

ans.push(6);

similarly 7

ans = [3, 3, 5, 5, 6, 7]