1248. Count Number of Nice Subarrays

Solved

Medium | Topics | Companies | Hint

Given an array of integers `nums` and an integer `k`. A continuous subarray is called **nice** if there are `k` odd numbers on it.

Return *the number of **nice** sub-arrays.*

Example 1:

```
Input: nums = [1,1,2,1,1], k = 3
Output: 2
Explanation: The only sub-arrays with 3 odd numbers are [1,1,2,1] and [1,2,1,1].
```

Example 2:

```
Input: nums = [2,4,6], k = 1
Output: 0
Explanation: There are no odd numbers in the array.
```

→ It is 3rd type of Striver window

```java
class Solution {
    public int numberOfSubarrays(int[] nums, int k) {
        return cntNum(nums , k)-cntNum(nums , k-1);
    }
    public static int cntNum(int[] nums , int k){
        int cnt=0 ,l=0 , r=0 ,odd=0, n=nums.length;
        while(r<n){
            if(nums[r]%2!=0){
                odd++;
            }
            while(odd>k){
                if(nums[l]%2!=0) odd--;
                l++;
            }
            if(odd<=k){
              cnt += (r - l + 1);
            }
            r++;
        }
        return cnt;
    }
}
```

Approach:-

number of subarray having number of odd digit is less than k

Concept related to formula (r-l+1)

1. r-l+1 (length of array b/w r, and l)
2. r-l+1 (No. of subarray having end 'r' as element Between l)

Example
0 1 2 3
a b c d e
l r

d
cd
bcd

{length = 3-1+1 = 3}

```java
class Solution {
    public int numberOfSubarrays(int[] nums, int k) {
        return cntNum(nums , k)-cntNum(nums , k-1);
    }
    public static int cntNum(int[] nums , int k){
        int cnt=0 ,l=0 , r=0 ,odd=0, n=nums.length;
        while(r<n){
            if(nums[r]%2!=0){
                odd++;
            }
            while(odd>k){
                if(nums[l]%2!=0) odd--;
                l++;
            }
            if(odd<=k){
              cnt += (r - l + 1);
            }
            r++;
        }
        return cnt;
    }
}
```

Approach :

: number of subarray having the sum is less than k

Example {1, 1, 2, 1, 1} k=3

→ {Remove odd Number to 1 (and) Remove even Number to 0}

→ {1, 1, 0, 1, 1} k=3

Now → Number of subarray having sum is less than (K<=3)

Note:

- Direct Array को change (or) new Array नहीं बनाते हैं।
- fetch करते हैं, और change करते हैं।