

<https://course.acciojob.com/idle?question=faddb193-7416-4426-8c42-99dcd35916a8>

**MEDIUM**

**Max Score: 40 Points**

## Subarray Product Less Than K

Given an array of integers `nums` and an integer `k`, return the number of contiguous subarrays where the product of all the elements in the subarray is strictly less than `k`.

### Input Format

First line contains an integer `n`.

Next line contains `n` spaced integers.

The last line contains `k`.

### Output Format

Return the number of contiguous subarrays with product of all elements in the subarray is strictly less than `k`

### Example 1

Input

```
4
10 5 2 5
100
```

Output

```
8
```

Explanation

The 8 subarrays that have product less than 100 are: [10], [5], [2], [6], [10, 5], [5, 2], [2, 6], [5, 2, 6]

Note that [10, 5, 2] is not included as the product of 100 is not strictly less than k.

## Example 2

Input

```
3
1 2 3
0
```

Output

```
0
```

Explanation

No subarray is possible with product less than K.

## Constraints

$1 \leq \text{nums.length} \leq 3 * 10^4$

$1 \leq \text{nums}[i] \leq 1000$

$0 \leq k \leq 10^6$

### Topic Tags

Sliding Window

Arrays

# My code

// in java

```
import java.util.*;
```

```
class Solution {  
    public int numSubarrayProductLessThanK(int[] nums, int k) {  
        //Write code here  
        int n=nums.length;  
        int prod=1;  
        int j=0;  
        int ans=0;  
        for(int i=0;i<n;i++)  
        {  
            prod*=nums[i];  
            while(prod>=k && j<i)  
                prod=prod/nums[j++];  
            if(prod<k)  
            {  
                int l=i-j+1;  
                ans=ans+l;  
            }  
        }  
        return ans;  
    }  
}
```

```
public class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        int n;  
        n = sc.nextInt();  
        int arr[] = new int[n];
```

```
    for (int i = 0; i < n; i++)  
        arr[i] = sc.nextInt();  
    int k;  
    k = sc.nextInt();  
    Solution Obj = new Solution();  
    int result = Obj.numSubarrayProductLessThanK(arr, k);  
    System.out.println(result);  
    sc.close();  
}  
}
```