

Count Number of consecutive 1's

problem: Given a array, count maximum number of consecutive 1's that can be achieved by flipping maximum k 0's.

solution Approach:

This questions belongs to variable size window. where we have to find longest subarray count (window size) , which can found by flipping max k zero's.

1. base case , if number of zero in the given array is less then given k i.e number of zero that can be flipped, then return array size as response, because all the zero's can be flipped.
2. then apply sliding window.
3. condition : number of zero equals to given k value
4. If number of zero grater then k, then sliding window from start point.
5. If number of zero = k, then find maxConsecutiveCount.

```
public int longestOnes(int[] nums, int k) {
    int i = 0, j = 0;
    int numZeroes = 0;
    int maxConsecutiveOnes = 0;
    for(int l=0; l< nums.length; l++){
        if(nums[l] == 0){
            numZeroes++;
        }
    }
    if(k > numZeroes){
        return nums.length;
    }
    numZeroes = 0;
    while(j < nums.length){

        if(nums[j] == 0){
            numZeroes++;
        }

        if(numZeroes == k){
            maxConsecutiveOnes = Math.max(maxConsecutiveOnes, j- i + 1);
        }
        else if(numZeroes > k){
            if(nums[i] == 0){
                numZeroes--;
            }
            i++;
        }
        j++;
    }
}
```

```
    return maxConsecutiveOnes;  
}
```