1 sec → $10^8 - 10^9$ iterations

$N <= 10^6$

$O(N)$ ✓

$O(N^2)$ ×

$O(N \log N)$ ✓

$10^6 \log 10^6$

$10^6 \log 10^3 \times 10^3$

$10^6 \log 2^{10} \times 2^{10}$

$10^6 \log 2^{20}$

20×10^6

$N <= 20$

~~$O(N)$~~

~~$O(N^2)$~~

~~$O(N^3)$~~

$O(2^N)$

$O(N!)$

$N <= 10^{10}$

$O(\log N)$

$O(\sqrt{N})$

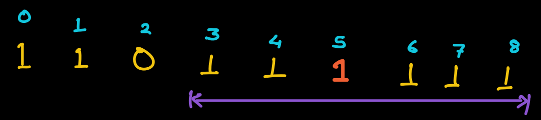
Q

Amazen
MS
Adobe
⋮

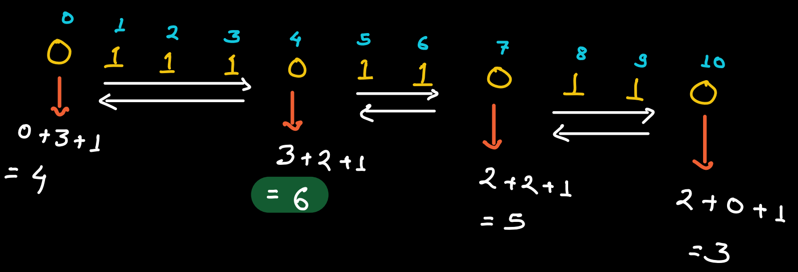
Given an array of 1 & 0. We can replace one of the 0 with a 1. Return the count of max consecutive 1's in the array.

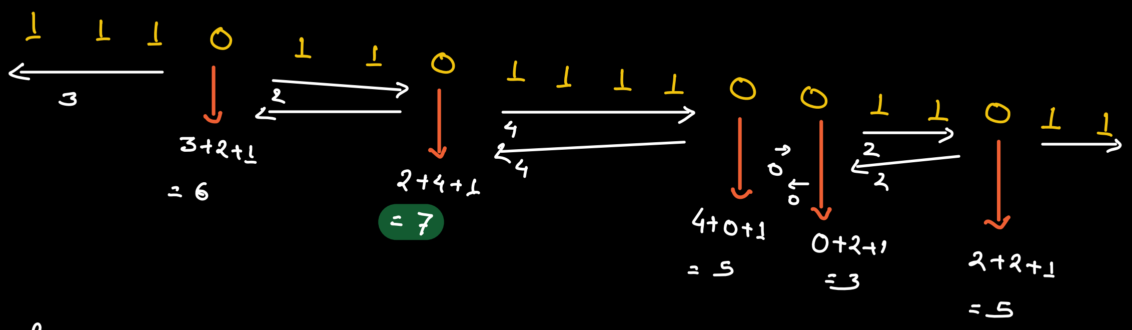
Ex :

0	1	2	3	4	5	6	7	8
1	1	0	1	1	0	1	1	1



Ex :



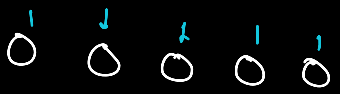


for every 0 in the array:

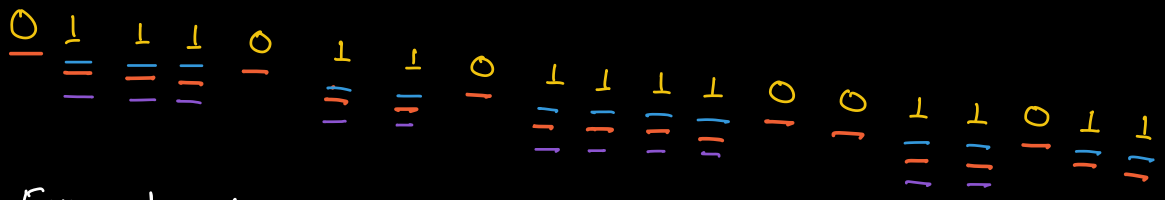
- Count number of consecutive 1's on left → l
- Count number of consecutive 1's on right → r
- if (l+r+1 > ans) { ans = l+r+1 }

Edge Case: If all are 1 (no 0's)

1 1 1 1 1 1



```
→ for ( i 0 → N) {
      if (A[i] == 0) {
          for (  →  ) → ①
          for (  →  )
      }
  }
```



Every element is getting accessed at max 3 times

∴ # iterations → 3N

TC : O(N)

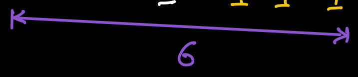
Amazon
Direct-i
MS
Adobe
⋮

Q Given an array of 1 & 0. We can swap one of the 0 with a 1. Return the count of max consecutive 1's in the array.

Ex :

0	1	2	3	4	5	6	7	8
1	1	0	1	1	0	1	1	1

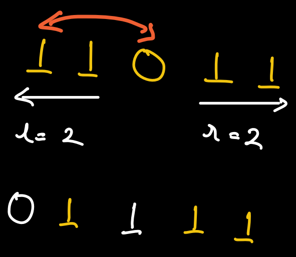
0	1	2	3	4	5	6	7	8
0	1	0	1	1	1	1	1	1



- Calculate total count of 1's
- for every 0 in the array:
    - Count number of consecutive 1's on left → l
    - Count number of consecutive 1's on right → r

$$\text{Count} = \begin{cases} l+r & \text{if } (l+r) == \text{total count of } 1 \\ l+r+1 & \text{if } (l+r) < \text{total count of } 1 \end{cases}$$

- if (count > ans) { ans = count }



total Count of 1's → 4

$(l + r) == 4$?

Amazon Q

VMware

No of triplets

Given an array. Count the number of triplets i, j & k such that

$$i < j < k$$

$$\& \ A[i] < A[j] < A[k]$$

A : $\overset{0}{2}, \overset{1}{6}, \overset{2}{9}, \overset{3}{4}, \overset{4}{10}$ ⟶ 5

i	j	k	A[i]	A[j]	A[k]
0	1	2	2	6	9
0	1	4	2	6	10
0	3	4	2	4	10
0	2	4	2	9	10
1	2	4	6	9	10

idea : iterate over all possible triplets

```
count = 0;
for ( i=0; i<N; i++ ){
    for ( j=i+1; j<N; j++){
        for ( k=j+1; k<N; k++){
            if ( A[i] < A[j] && A[j] < A[k])
                count ++;
            }
        }
    }
}
```

TC: $O(N^3)$

	0	1	2	3	4	5	6
A :	4,	1,	2,	6,	9,	7,	2

Hint : In how many triplets index 3 is in middle

$$\underset{i}{\underline{\quad}} < \underset{j}{\underline{\ 3\ }} < \underset{k}{\underline{\quad}}$$

left		right
	6	
4		9
1		7
2		

triplets = 3×2
= 6

No of triplets in which any A[x] will be in middle :

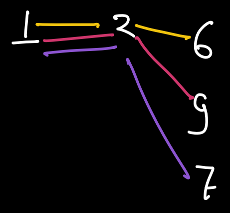
No of elements smaller than A[x] on left of x

×

No of elements greater than A[x] on right of x

	0	1	2	3	4	5
	4	1	2	6	9	7
l	0	0	1	3	4	4
r	3	4	3	2	0	0
	0	0	3	6	0	0

→ 9 (Summation of all)



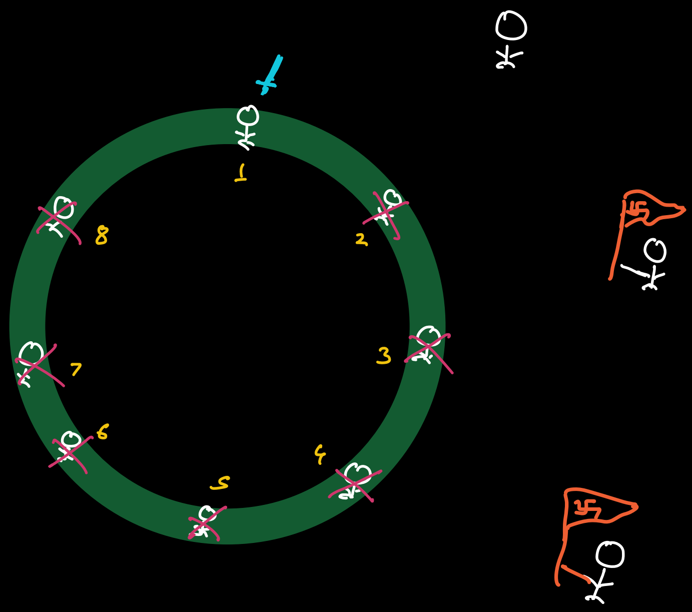
for evey element A[i] ; ⟶ N iterations

(N-1) iterations

- iterate from index 0 to i-1 & count the elements smaller than A[i] ⟶ l
- iterate from index i+1 to N-1 & count the elements greater than A[i] ⟶ r
- ans = ans + $l \times r$

TC : $O(N^2)$

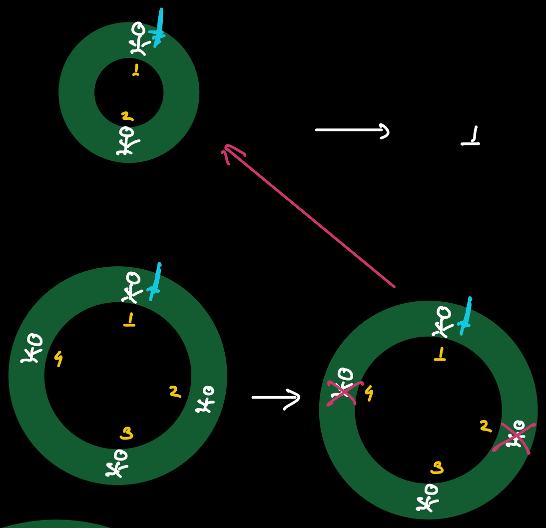
Josephus Problem



N = 1

⟶ 1

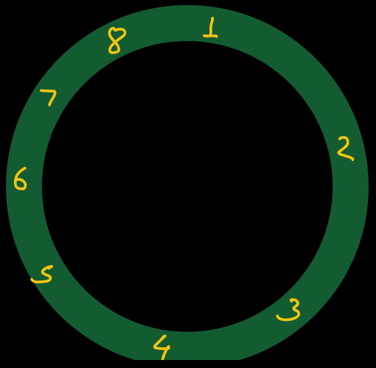
N = 2



⟶ 1

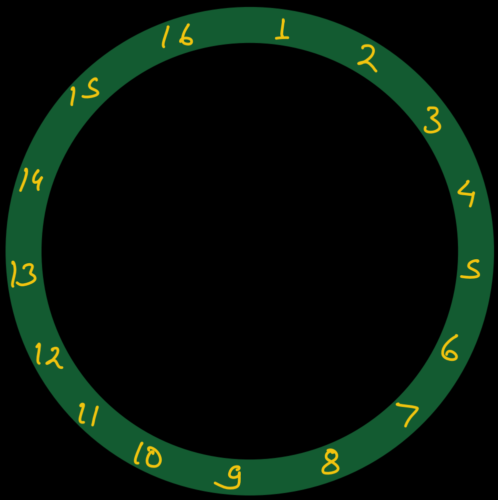
N = 4

N = 8



⟶ 1

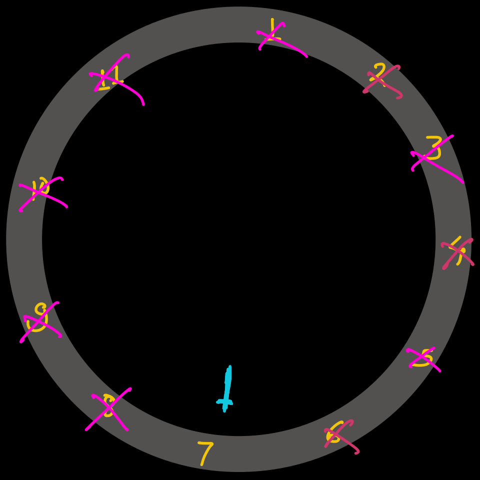
N = 16

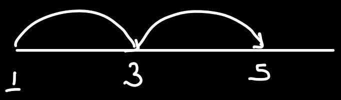


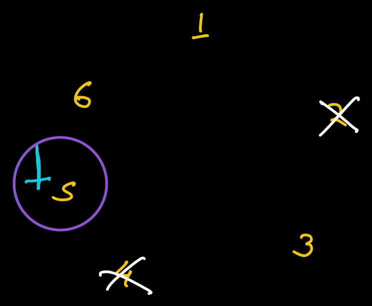
If N is a power of 2 ⟶ 1 always win in that case

(Who ever starts killing)

11 ⟶ 10 ⟶ 9 ⟶ 8







$6 \rightarrow 5 \rightarrow 4$

$N = 100 \xrightarrow{36\ \text{kill}} 64$

$1 + 36 \times 2$

$\Rightarrow 73$

$N = 11 \xrightarrow{3\ \text{Kill}} 8$

$1 + 2 \times 3$

$\Rightarrow 7$