राधा राधा

Additional Problems

Q1 Minimum Swaps

A = [1 , 12 , 10 , 3 , 14 , 10 , 5]

B = 8

⇒ elements less than 8 = 1, 3, 5 = 3 elements

∴ window size = 3

⇒ the window with the least number of bad Numbers (A[i] > B) is the no. of swaps needed

Q2 Row with maximum number of 1's

A =

0	1	1
0	0	1
0	1	1

Observation

→ Since the columns are sorted row wise, the 1's will be present at the end of each row

→ Hence, we can start traversal from top-right corner of the matrix

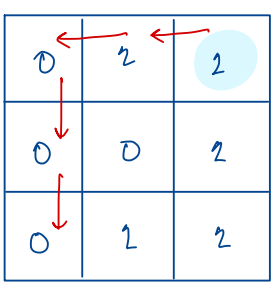
→ we can keep on going left until 0 is found, store no of 1's for that row

→ As soon as 0 is found, go to next row, if element present there is 2, start going left in that row and storing number of 2's (we do not need to check right as since row is sorted, it is obvious to the right 1's are present, y which we already have the count from previous row)

→ If the element found at next row is 0, keep going down until either 2 is found, or we have exhausted boundary

Code

A =




```
i = 0, j = M-1, index = N, count = 0;
while (i < N && j >= 0)
    int ele = A[i][j]
    if (ele == 2) {
    | count++; j--; rowIdx = i;
    } else {
    | i++;
    }
return rowIdx;
```