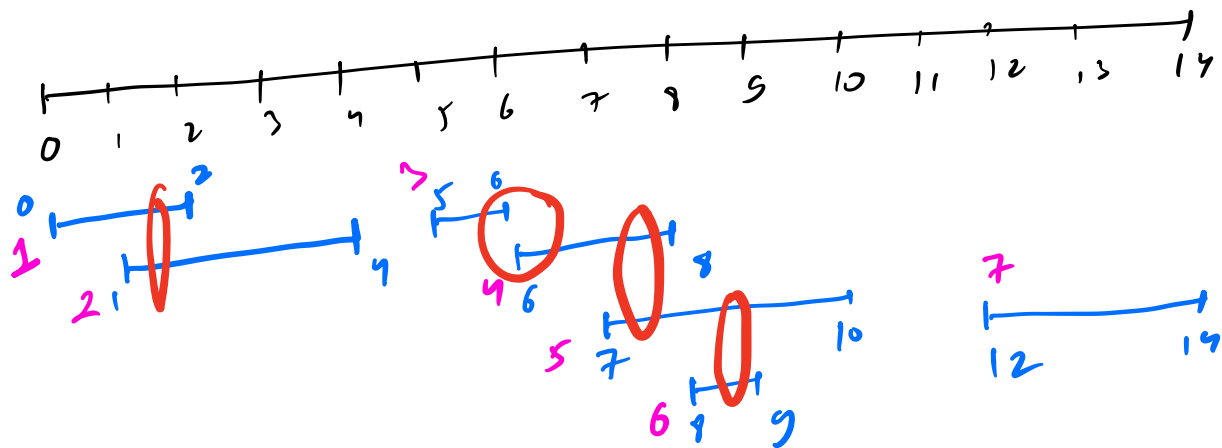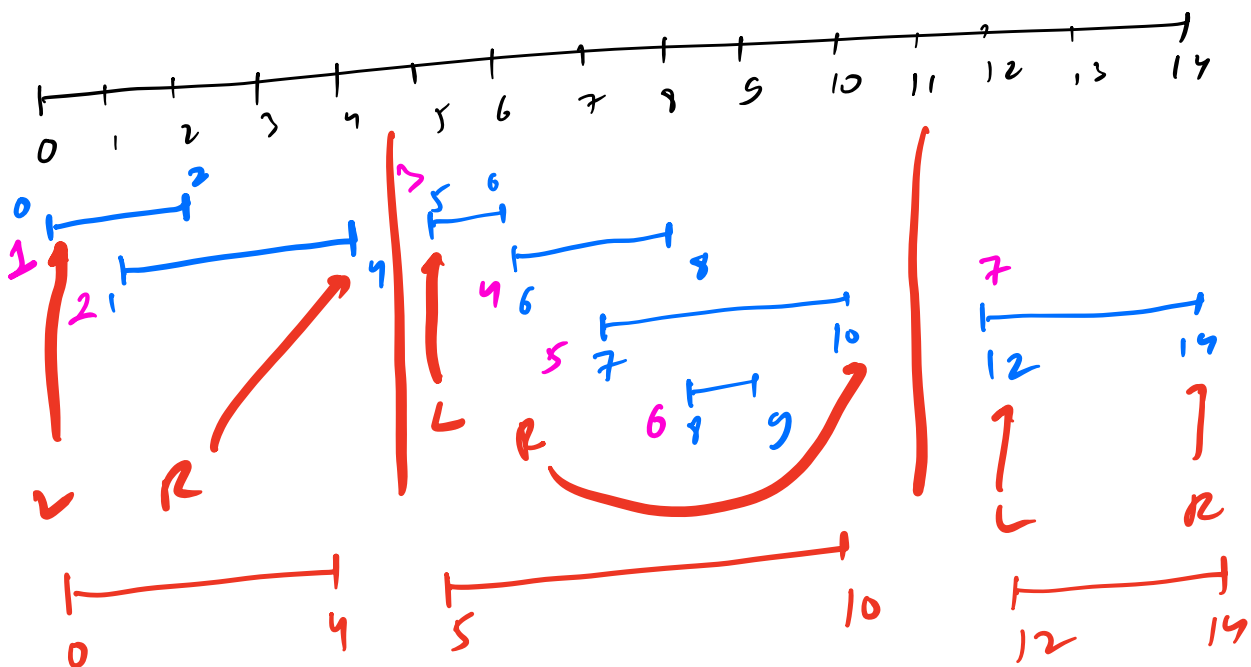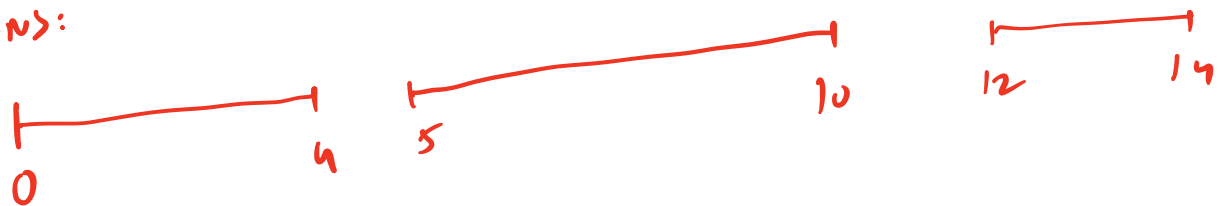**Q** Given a sorted list of overlapping intervals, sorted based on start time. Merge all the overlapping intervals & return the sorted list. **S[], E[]**
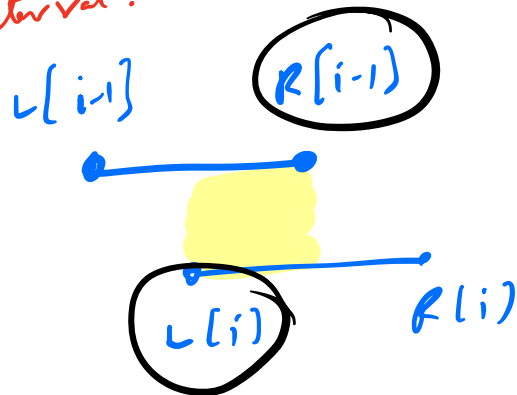

ANS:

→ Adjacent intervals would overlap to form a new interval.

→ Condition for i^{th} interval overlapping with $(i-1)^{th}$ interval.

$L[i-1]$ $R[i-1]$

if $(L[i] <= R[i-1])$
→ overlap!

$L[i]$ $R[i]$

```
// S[], E[], N

L = S[0] R = E[0];

f ( i = 1; i < N; i++) {
    if ( S[i] <= R ) {
        R = max(R, E[i]);
    }
    else {
        print(L, R);
        L = S[i], R = E[i];
    }
}
print(L, R);
```
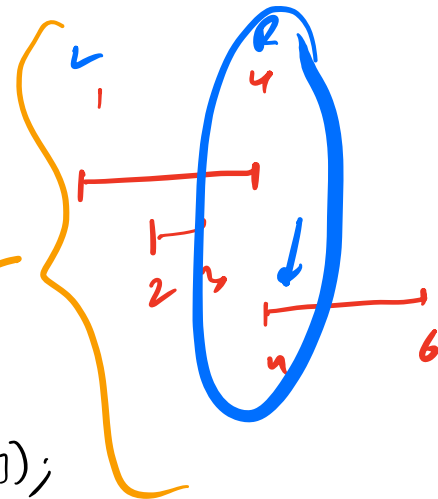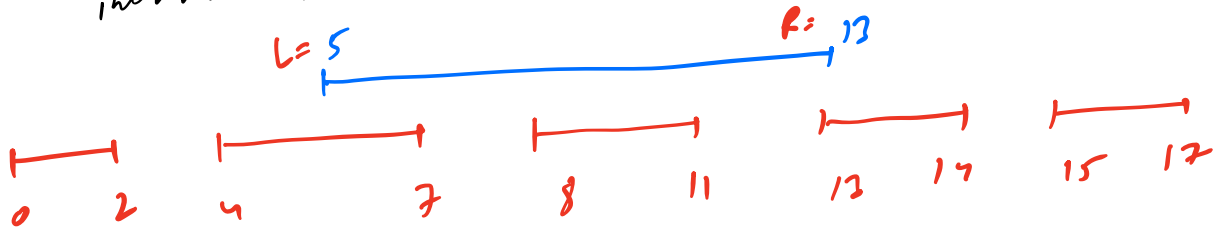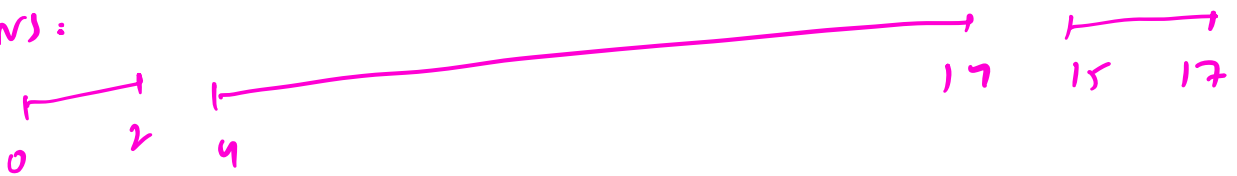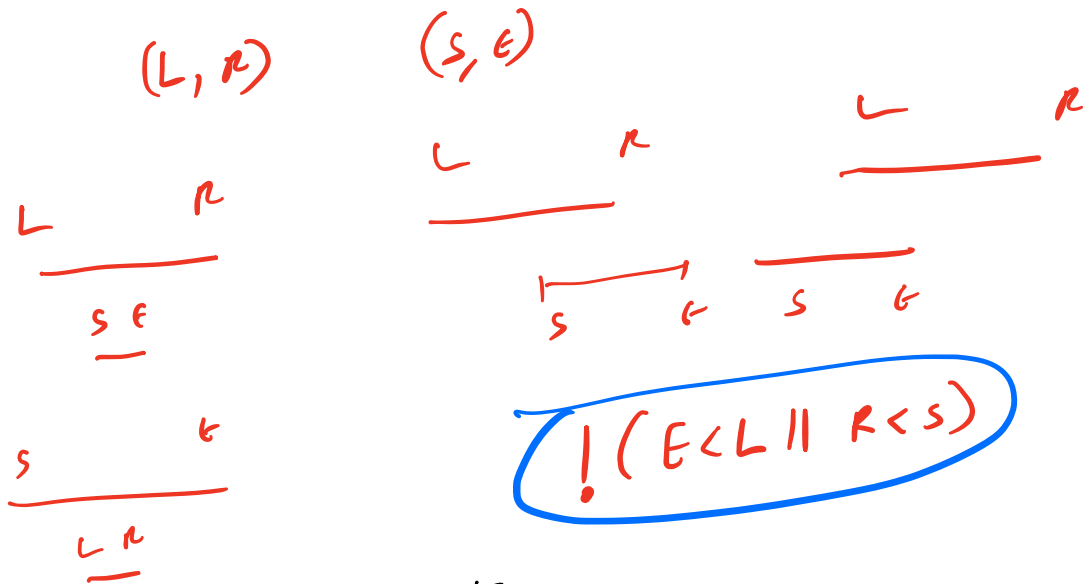
TC = O(N)

SC = O(1)

Q. Given a sorted set of non-overlapping intervals.
Insert a new interval s.t. the final list of
intervals is also sorted & non-overlapping.

L = 5   R = 13

0   2   4   7   8   11   13   14   15   17

ANS:

0   2   4   14   15   17

Condition for over lapping →

(L, R) (S, E)

L   R   L   R   L   R

     S E   S   E   S   E

S   E

     L R

→ Non-overlap condition →

S   E   L   R   S   E
     L   R   or

                         $(E < L \ || \ R < S)$

$!(E < L \ || \ R < S)$

L R

$S[i]$ $E[i]$ $S[j]$ $E[j]$

0 $i-1$ $j+1$ $N-1$

$i \; .. \; i+1 \; \text{———} \; j-1 .. \; j$

1st ov.
interval

last ov.
interval
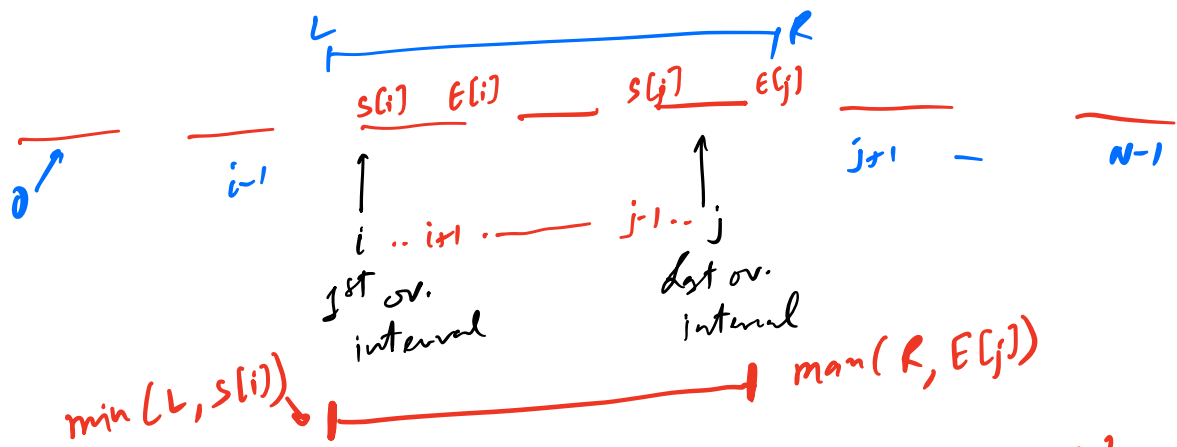
$\min(L, S[i])$ $\max(R, E[j])$

ANS: $\{ I_0, I_1 . \text{———} I_{i-1}, I\{\min(L, S[i]), \max(R, E[j])\},$

$\qquad I_{j+1}, I_{j+2} \; \wedge \; \text{——} , I_{N-1} \}$

Corner Case:

L R
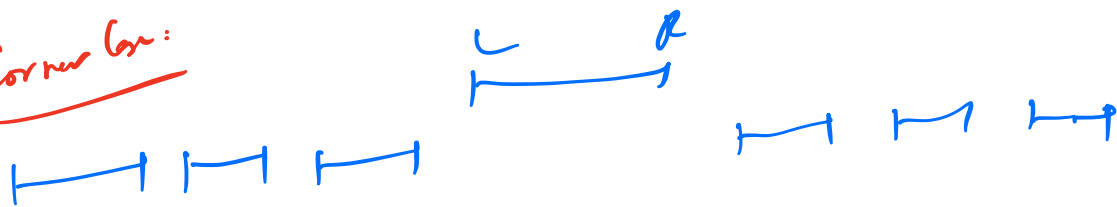
TC: $O(N)$

SC: $O(1)$

Q Given a 2D array.
Find the MAX sub-matrix sum.

$$A = \begin{bmatrix} 1 & 2 & 1 & 4 & 2 \\ 0 & -1 & 2 & 3 & 1 \\ 0 & -5 & -2 & 1 & 2 \\ -3 & 1 & 3 & 2 & 1 \end{bmatrix} \rightarrow 20$$

$$N \begin{array}{c} m \\ \boxed{} \end{array}$$
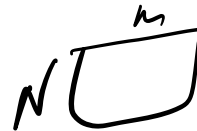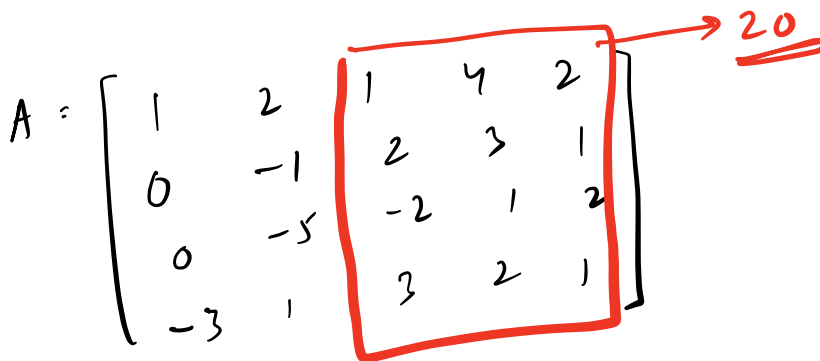
$$\rightarrow N^2 M^2$$

I)

Consider ALL SMs

$\rightarrow$ find their sum
& take MAX $\Bigg\}$ $\rightarrow$

BF	PS
NM	1

$$\boxed{TC = O(N^2 M^2)}$$

# of Combination
of green lines
$= N^2$

M

N

T
B

N

X
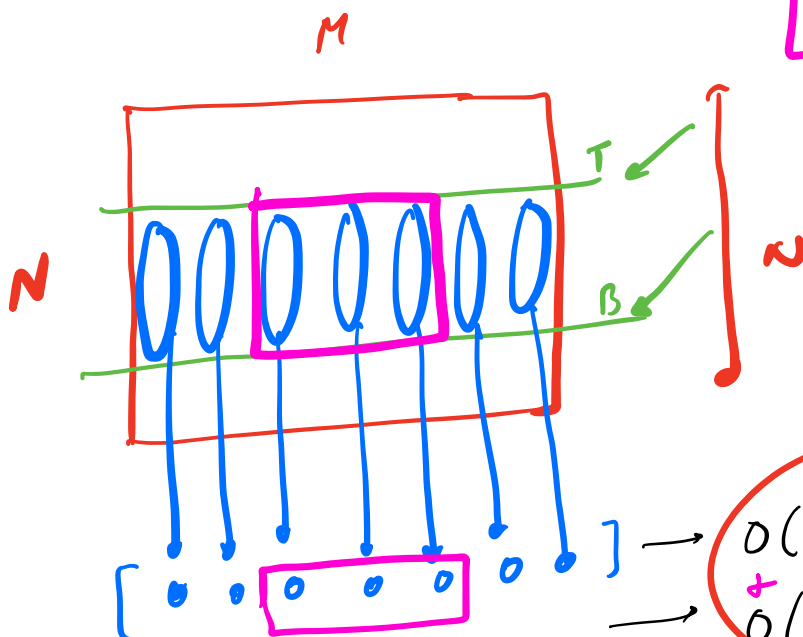
# of Combination
of green lines $= N^2$

$O(M)$ [generating]
+
$O(m)$ KADANE

$$\boxed{TC : O(N^2 M)}$$

$$\boxed{\begin{array}{c} SC \\ = O(NM) \end{array}}$$
$O(1)$

1. Create **PS** colwise

2.
```
f ( T = 0; T < N; T++) {
  f ( B = T; B < N; B++) {
    // Green line [T, B)
    Ar []
    f ( j = 0; j < M; j++) {
      Ar [j] = PS [B] [j] - PS [T-1] [j];
    }
    ANS = max (ANS, KADANE ( Ar [] ) );
  }
}
```
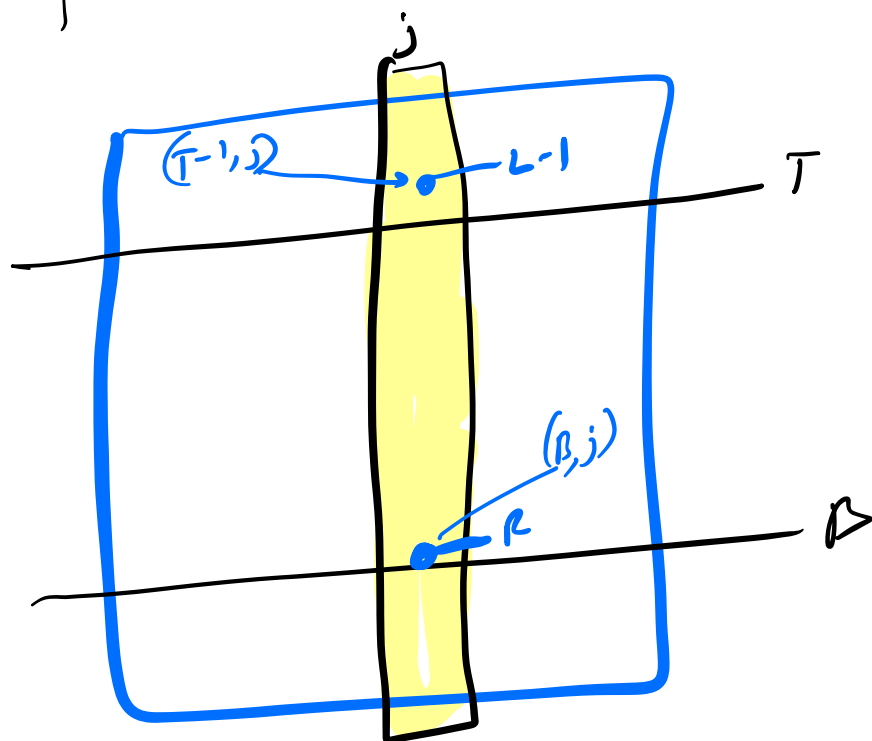
$$TC = O(N^2 M)$$

$N >> M$

$\boxed{NM} + M^2 N$

transpose

$M << N$

II

$M < 10$

$N > 1000$

Row wise PS

M^2

$\times$

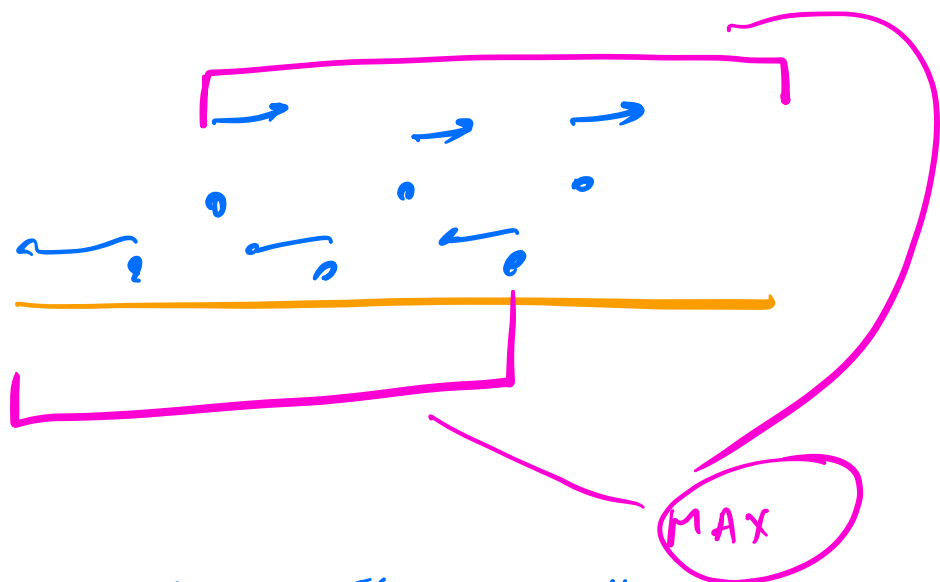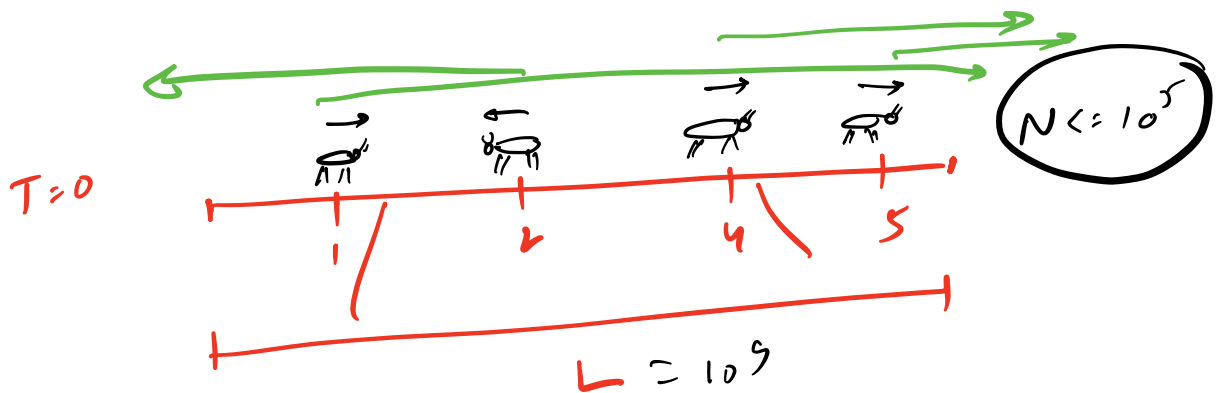$O(N)$

$O(M^2 N)$

$$TC = O(\min(N, M)^2 \cdot \text{Max}(N, M))$$

$$SC = O(NM)$$

$O(1)$ SAME

1
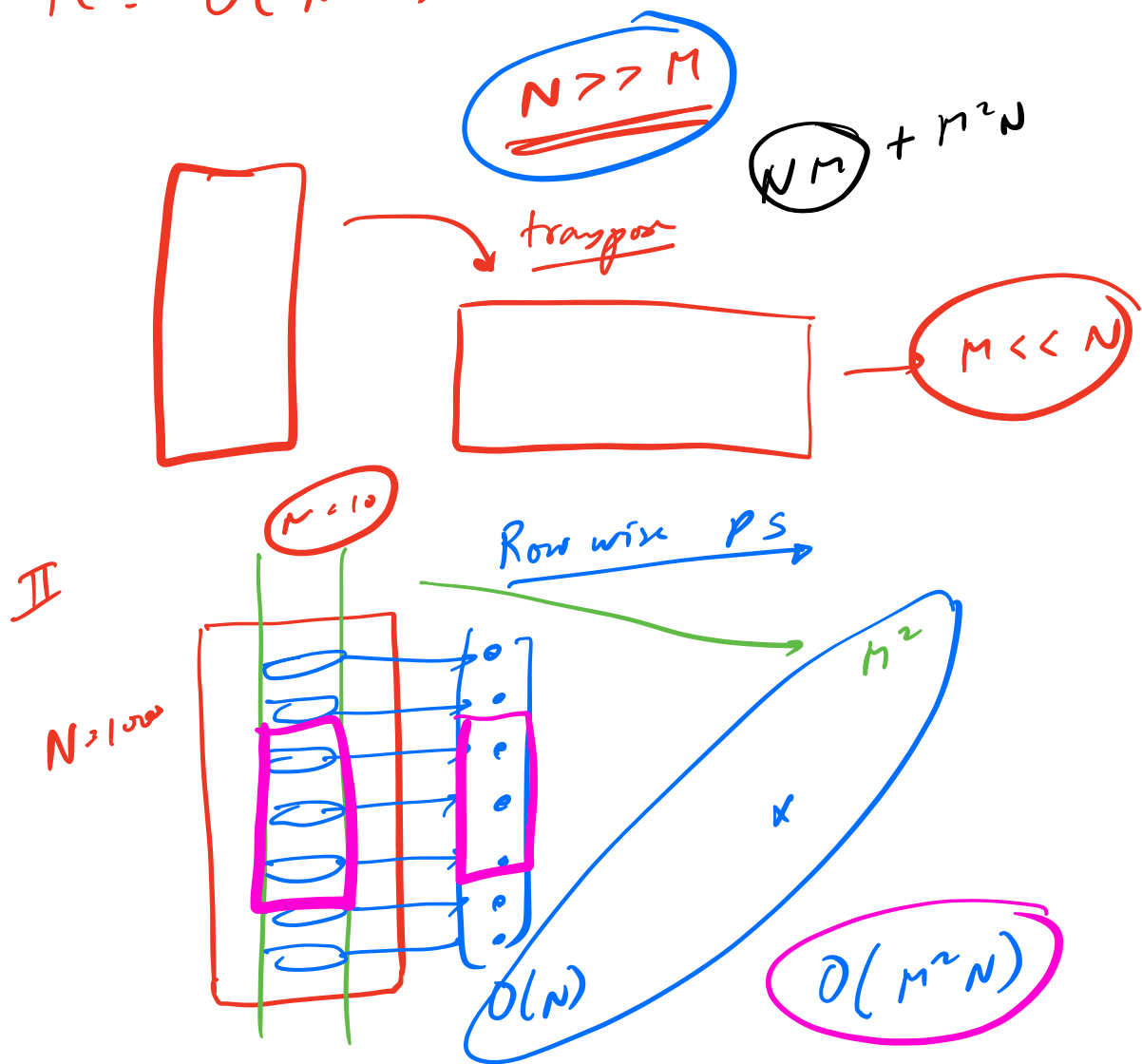
Given a table of length L.
N ants on it at DISTINCT positions.
Their pos & direction is given.

→ If 2 ants collide head on, they turn around!

→ find the time at which the last ANT falls off the table!

0.5 s

$N <= 10^5$

T=0

1   2   4   5

$L = 10^9$

Idea: Imagine the ANTS passing through!
→ SAME as colliding & turning back!