

Search in 2D Matrix I

| | | | |
|----|----|----|----|
| 1 | 3 | 5 | 7 |
| 10 | 11 | 16 | 20 |
| 23 | 30 | 34 | 60 |

target = 34

- each row is sorted in non-decreasing order
- the first integer of each row is greater than the last integer of the previous row

we can do linear search $\Rightarrow TC = O(n \times m)$
but we want in $O(\log(m \times n))$

\therefore we can apply **binary search**

now for each row ka jo range hai
(st, end) will be unique
and compare with target
and the range is non overlapping / unique range

① Search correct row
BS \rightarrow rows

$$sr = 0, er = m - 1$$

$$midR = sr + (er - sr) / 2$$

① $midR \Rightarrow$ target exists

$$mat[midR][0] = < tar <= mat[midR][n-1]$$

② $tar > mat[midR][n-1]$

$$sr = mid + 1$$

③ $tar < mat[midR][0]$

$$er = mid - 1$$

② After finding correct row, now search correct column

BS \rightarrow columns

$$st = mat[midR][0]$$

$$end = mat[midR][n-1]$$

last element
in row
 $\sqrt{20}$