ELEMENT SEARCH IN A 2D MATRIX

✡ In this problem, we are given a 2D matrix of integers and we are supposed to return the row, column of that number if it exists

Bruteforce solution is obviously to run a M x N loop and find the element. Time Complexity is O(MN)

Better solution is to find which row will have the element and then binary search in that row. Time complexity is O(M + log N)

Optimal solution is to flatten out the entire array as it is fully sorted array and just run a binary search. Time Complexity is O(log MN)

Pseudocode :

```
eleSearch(arr, M, N, K) {
    total = N * M - 1
    ele = (-1, -1)
    low = 0
    while (low <= total)
        mid = (low + total) / 2
        i = mid / M
```

```
        j = mid % N
        if(arr[i][j] == K) {
            ele = (i, j)
            return ele
        } else if (arr[i][j] < K) {
            low = mid + 1
        } else {
            high = mid - 1
        }
    }
    return ele
}
```