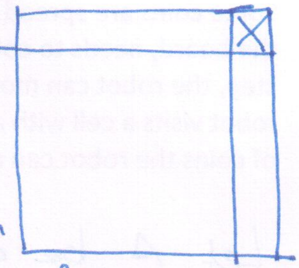
n^2 different numbers are written on n^2 cards, one number per card. The cards are arranged in n rows and n columns, in increasing order in each row (left to right) and each column (top down). All the cards are turned faced down so that you cannot see the numbers written on them. Devise an algorithm to determine whether a given number is written on one of the cards by turning up less than $2n$ cards? (10)



Solution: Consider the element at location $(1, n)$, say this is p.

Suppose we are searching for x.

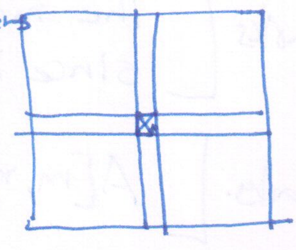
If $x < p$ then x cannot be in the last column since all elements there are larger than p. In this case eliminate last column.

If $x > p$ then x cannot be in the first row since all elements in 1^{st} row are less than p. Eliminate first row.

If $x = p$ done. STOP.

By opening one card we eliminate either one row or one column. Hence only $2n$ cards need to be checked.

Common mistakes: If you open the middle element $(\frac{n}{2}, \frac{n}{2})$ then you will still have to solve three subproblems in arrays of size $n/2 \times n/2$. So the recurrence you get will be



$$T(n) = 3T(\tfrac{n}{2}) + 1$$

whose solution is $n^{\log_2 3}$

Marking scheme:
- 2 marks for an $O(n \log_2 n)$ solution
- 2 marks if you opened the $(n/2, \frac{n}{2})$ element & said that you need to solve 3 subproblems of size $n/2 \times n/2$.