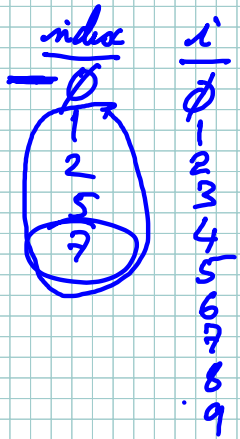


arr

0	1	2	3	4	5	6	7	8	9
2	15	22	5	17	83	74	16	42	1



```

int findMax(arr){
    int index = 0;
    for (i = 0; i < 10; i++) {
        if (arr[i] > arr[index]) {
            index = i;
        }
    }
    return index;
}

```

$O(n)$

	0	1	2
0	72	55	80
1	2	95	15
2	17	22	99

$n=3$

$O(n^2)$

<u>maxI</u>	<u>maxJ</u>	<u>i</u>	<u>j</u>
0	0	0	0
0	2	1	1
1	1	1	0
2	2	1	1
		2	0
		2	1
		2	2

```

int findMax(arr) {
    int maxI = 0;
    int maxJ = 0;
    for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
            if (arr[i][j] > arr[maxI][maxJ]) {
                maxI = i;
                maxJ = j;
            }
        }
    }
    return arr[maxI][maxJ];
}

```