Computer Programming (CS0.101)

[Monsoon 2021-22]

Coding Quiz - Mid Semester Exam

---

2 Questions - 100 marks Each => 200 Marks

Q1 . Given the distribution of people in each rooom of a hotel consisting of 100 floors and 30 rooms on each floor, print the no of rooms consisting of 0 people, 1 people, ... 8 people.

Input Format

```
a_1_1 a_1_2 a_1_3 ............. a_1_30
a_2_1 a_2_2 a_2_3 ............. a_2_30
    ............................
    ............................
a_100_1 a_100_2 a_100_3 ....... a_100_30
```

Output Format

```
cnt_0
cnt_1
cnt_2
..
..
..
cnt_8
```

Sample Test Case

INPUT ->

```
6 2 1 2 3 4 6 8 0 4 5 7 7 7 6 3 5 4 5 6 0 4 0 6 5 7 2 1 2 0
6 3 1 6 1 1 1 2 5 2 5 6 0 1 5 1 2 3 4 3 7 2 2 2 5 2 5 3 1 2
0 3 1 1 8 4 8 3 6 6 0 0 1 1 3 6 4 7 7 0 8 7 5 5 3 4 4 0 8 8
5 5 1 2 8 3 1 4 4 8 1 5 4 8 6 4 0 6 4 4 0 2 3 4 5 5 3 5 1 7
5 2 1 1 2 7 6 3 7 0 7 3 7 8 3 3 7 2 5 0 8 5 5 6 6 8 5 0 1 1
3 1 2 0 5 4 8 7 8 4 8 8 6 2 1 0 0 5 2 1 0 1 3 1 2 0 5 1 5 8
1 1 8 1 0 2 1 6 3 6 6 6 4 7 5 3 0 0 2 4 5 1 5 4 2 2 6 8 7 6
3 0 3 4 7 3 5 4 4 0 0 1 0 8 6 5 7 4 8 2 4 2 3 8 1 7 7 8 2 4
8 4 1 4 8 2 6 3 0 2 0 3 5 5 7 2 4 6 7 4 2 6 0 2 0 3 5 8 6 3
1 2 5 4 8 8 4 3 1 3 6 2 7 5 5 7 6 8 5 7 0 1 7 3 8 3 6 3 1 7
5 2 7 4 6 6 5 7 6 1 3 3 6 4 8 0 0 0 1 5 7 5 1 8 4 5 1 4 2 8
1 3 4 1 2 4 6 7 7 6 3 6 7 2 7 7 7 1 5 2 1 6 5 0 7 0 8 1 8 0
0 7 2 7 1 6 6 4 1 7 5 7 5 5 8 1 5 5 2 0 1 2 6 8 0 0 0 2 2 2
1 0 3 6 0 0 4 3 2 6 5 7 3 4 8 0 1 2 3 5 3 7 1 1 2 2 6 6 4 6
4 8 6 3 6 5 4 0 8 6 2 6 8 1 1 1 0 3 4 0 8 1 8 4 2 7 6 6 5 6
6 0 5 1 6 3 0 8 4 3 1 4 7 6 1 7 4 0 0 3 1 4 0 0 6 8 7 0 2 4
7 0 2 7 6 8 1 0 8 2 6 1 5 2 0 0 1 5 4 5 3 7 8 4 3 7 4 8 7 0
7 4 2 0 0 7 6 7 3 3 3 8 7 0 8 6 3 0 5 4 4 1 6 5 4 3 5 7 4 3
```

4 4 5 0 8 4 0 6 2 1 8 5 1 6 4 7 5 1 1 8 4 0 0 7 5 6 5 5 2 1
2 3 3 0 4 3 2 8 7 3 2 2 4 3 2 6 6 7 1 3 4 5 5 2 3 4 6 8 2 1
6 8 1 8 5 2 1 4 3 6 5 1 4 5 2 1 6 0 1 5 8 2 5 6 0 1 2 3 8 7
2 1 1 7 4 3 2 0 2 5 8 2 8 6 5 4 7 6 6 6 4 0 0 3 6 0 2 7 1 1
3 6 4 3 8 1 4 1 5 4 3 1 6 8 0 5 6 4 1 4 7 3 5 1 7 4 1 4 6 6
3 3 1 4 2 2 6 3 7 6 6 0 2 8 5 0 0 1 4 1 5 0 0 1 5 1 6 5 6 3
1 7 0 2 5 5 3 4 0 8 6 5 8 8 5 5 7 5 1 1 1 4 2 6 5 7 8 6 2 1
7 7 8 6 1 0 8 1 8 0 6 5 7 7 8 7 7 3 1 5 2 1 8 0 4 7 3 4 1 1
1 2 5 4 5 4 8 6 5 6 1 3 6 1 3 2 8 4 0 4 7 2 5 8 4 1 6 6 0 4
8 8 2 3 0 7 5 6 1 5 6 3 2 7 1 0 2 3 7 3 2 2 0 2 8 6 3 3 2 7
8 0 7 1 8 8 4 2 8 4 4 4 3 6 0 2 0 5 3 4 1 2 0 6 1 6 0 4 4 3
4 3 8 2 5 2 3 2 2 1 6 5 4 3 1 2 4 1 5 6 2 3 2 2 0 5 5 4 5 6
6 2 4 6 2 3 0 2 5 5 4 4 0 3 4 0 3 4 4 5 7 1 5 8 4 2 0 4 2 5
8 3 5 6 7 1 0 6 7 8 3 0 2 5 8 3 0 4 0 3 5 1 6 7 7 3 3 7 3 4
1 3 5 0 2 8 4 5 5 6 4 4 2 8 8 3 7 2 5 7 1 1 1 4 6 8 1 8 2 2
8 6 8 8 0 0 7 2 7 2 8 1 1 2 5 4 7 8 8 1 0 4 1 7 3 1 0 8 1 8
2 4 3 2 4 8 0 7 4 4 5 0 1 3 1 7 8 3 2 1 8 5 2 2 3 4 8 4 5 3
4 8 2 4 2 8 1 0 6 2 3 4 2 8 2 7 5 1 6 6 3 2 7 4 0 1 7 1 6 7
3 5 0 2 6 1 6 3 0 5 8 6 0 3 4 4 0 2 1 8 2 2 4 0 4 2 5 3 2 6
8 1 2 1 3 5 5 4 2 4 1 4 8 3 7 0 0 2 2 4 5 0 6 5 1 1 5 5 2 0
3 2 0 7 7 0 5 0 4 6 7 4 4 8 0 5 5 3 7 2 2 6 3 0 1 3 0 3 6 3
2 2 4 0 2 2 2 0 6 8 7 4 7 6 6 3 1 8 3 6 6 8 4 3 5 5 2 8 3 6
0 0 8 6 4 8 2 8 3 3 4 6 2 5 6 7 4 3 8 7 4 4 6 2 1 7 0 0 7 0
2 3 7 0 8 7 8 8 3 7 5 6 7 5 3 3 0 5 0 1 3 5 5 5 2 8 3 5 0 0
3 8 3 6 8 8 2 2 8 0 7 2 4 8 0 4 6 3 7 1 5 4 3 6 3 3 5 1 4 1
4 1 7 3 6 8 5 1 1 3 4 7 5 3 8 4 3 4 1 1 7 1 5 8 7 8 0 5 4 3
0 6 8 5 0 8 7 4 1 5 0 6 7 7 8 6 4 3 4 2 8 2 6 5 5 5 4 2 4 3
7 8 7 8 2 8 6 2 6 8 3 4 5 8 6 6 3 6 7 1 7 3 2 4 3 6 7 6 8 3
2 4 2 2 8 5 1 5 2 5 1 2 1 5 3 5 2 5 6 2 1 8 2 0 5 1 5 4 5 7
5 5 2 1 0 5 7 8 0 8 6 7 2 8 1 2 1 4 2 8 7 3 3 2 1 6 7 5 6 0
0 4 5 4 6 7 7 8 1 7 6 7 4 2 2 4 3 8 0 7 6 5 8 8 1 6 8 3 2 0
7 1 4 1 4 6 7 4 4 8 2 6 2 7 2 1 3 8 5 4 1 0 8 5 8 1 8 8 6 2
2 7 1 1 4 0 2 7 4 5 6 8 3 7 2 4 6 7 3 5 0 2 7 6 5 0 7 0 5 4
4 6 4 7 2 8 6 1 3 7 5 7 4 1 6 2 7 7 6 4 3 4 4 1 3 2 1 2 4 6
3 2 3 5 7 0 8 0 2 1 2 2 4 3 7 0 3 6 5 1 8 3 3 4 8 8 5 5 3 8
7 1 3 5 2 4 5 7 8 8 3 2 6 5 0 3 2 8 6 8 8 3 0 4 4 3 8 2 8 0
4 3 3 1 7 3 0 7 0 3 8 3 1 1 3 3 8 8 6 2 4 6 1 6 0 8 2 0 3 2
8 7 8 8 7 4 2 3 7 4 1 3 3 6 8 1 4 3 6 2 6 4 0 3 7 4 7 8 0 1
5 0 3 8 8 8 6 6 2 6 1 3 5 7 6 5 0 6 0 0 3 5 6 4 4 7 0 7 0 6
3 8 1 5 3 0 8 6 1 2 8 6 8 2 2 2 6 8 8 3 3 7 2 0 4 4 0 8 7 6
5 6 0 7 7 4 8 7 8 8 5 4 3 8 8 6 3 5 6 5 4 7 6 0 4 5 0 5 4 4
0 4 6 2 5 6 7 3 2 3 7 6 5 5 5 3 5 7 1 2 3 4 1 5 4 7 3 0 1 2
6 2 6 4 0 2 2 5 8 0 6 0 1 0 1 6 8 1 8 3 4 8 8 3 2 4 5 0 7 8
8 4 7 1 5 8 4 3 0 5 1 6 3 5 1 6 8 0 4 1 0 1 4 0 6 4 4 0 2 5
8 6 6 4 2 4 5 7 1 2 8 0 4 0 3 6 4 0 3 2 7 6 1 6 3 4 7 3 3 5
0 0 3 5 7 5 4 8 0 0 2 0 7 0 3 4 7 2 8 6 0 7 4 3 2 3 6 5 7 5
6 6 2 4 7 0 1 3 5 6 6 5 8 7 4 4 1 5 2 8 2 0 8 5 5 0 1 8 1 2
6 6 0 8 7 3 0 5 5 4 7 3 8 2 1 5 8 5 1 6 3 0 1 0 4 8 5 7 1 5
6 5 3 4 3 6 0 6 4 1 3 7 6 5 2 3 4 1 3 1 7 4 5 0 6 0 1 0 6 7
5 5 0 1 7 0 1 0 3 4 0 2 6 0 7 1 0 4 8 7 3 8 0 1 1 2 8 6 8 5
5 7 4 1 3 1 7 3 6 1 1 3 2 3 6 3 0 3 2 6 4 3 2 5 2 5 6 3 8 5
0 8 3 2 8 6 0 4 8 0 8 1 6 5 8 4 6 6 1 3 6 0 0 1 7 0 1 4 2 5

```
2 1 0 3 7 5 2 0 3 5 2 2 8 0 6 4 2 1 8 5 7 6 3 4 6 3 2 1 6 4
6 2 5 5 2 8 8 1 1 8 2 2 0 6 8 7 5 6 4 7 7 2 2 1 6 2 2 2 8 7
7 3 1 6 5 5 5 0 3 2 2 8 7 1 5 5 2 7 6 1 1 5 7 8 6 1 5 1 1 8
7 6 7 5 4 7 2 6 5 8 6 7 0 8 7 3 6 4 3 1 3 1 0 4 0 7 7 4 7 4
2 4 2 7 3 4 8 8 7 7 8 1 0 6 1 6 6 7 3 7 6 8 7 1 2 8 7 7 8 5
3 4 8 6 2 1 7 0 2 4 2 2 8 1 3 7 6 3 1 6 5 3 0 6 0 1 4 2 0 2
2 6 8 1 8 3 5 8 3 7 3 7 7 8 3 5 0 5 3 1 2 4 4 8 6 5 6 8 6 0
7 5 7 7 7 8 3 1 6 7 0 4 7 0 4 5 4 7 1 4 1 5 1 7 2 5 2 4 0 4
0 0 2 7 2 2 8 6 7 8 5 1 4 6 7 8 1 3 8 0 2 4 7 6 0 2 2 0 1 4
1 8 1 7 6 1 3 8 6 5 7 8 6 8 5 7 2 8 7 3 7 8 6 5 6 8 1 7 8 2
7 2 8 6 6 0 1 3 5 2 4 1 1 4 8 6 6 5 4 4 1 1 2 1 7 6 4 5 5 7
2 5 5 4 2 8 7 8 3 4 4 6 6 7 4 8 8 0 0 8 3 8 3 7 2 1 6 1 3 4
8 2 4 2 6 0 5 3 8 0 8 1 3 3 3 1 7 1 4 8 5 5 7 2 7 4 6 1 5 7
8 5 2 2 0 3 8 6 0 1 8 5 0 1 2 3 2 8 4 3 8 7 5 3 8 3 5 4 5 3
3 8 5 4 0 6 6 2 7 5 2 4 1 3 8 1 8 7 0 1 2 4 0 8 8 4 2 7 5 7
2 2 8 7 2 5 6 0 4 6 0 7 2 8 1 8 1 2 6 8 0 0 2 6 0 7 4 8 6 4
4 6 6 7 0 6 0 2 0 0 7 1 6 1 0 4 6 6 1 8 8 1 3 4 1 6 6 1 4 4
7 1 0 4 3 1 3 6 6 1 8 8 6 2 8 3 0 2 0 8 3 0 1 7 8 8 6 4 1 5
8 6 3 3 1 6 3 0 6 0 5 4 4 8 0 2 8 8 6 6 5 7 7 7 8 7 7 3 0 6
5 5 2 4 0 8 7 8 2 0 5 3 5 5 1 7 3 5 8 0 8 4 6 6 1 6 6 8 3 5
0 2 3 2 6 2 0 4 8 6 8 4 8 1 3 5 6 8 4 4 2 4 1 7 5 8 0 4 0 0
4 3 5 1 3 7 6 1 3 3 7 8 0 3 5 0 5 6 7 8 3 3 1 3 2 0 1 5 2 4
3 4 5 7 0 7 3 0 3 2 7 2 0 5 3 1 0 3 8 7 1 0 6 2 8 5 4 0 4 2
8 4 3 0 2 0 4 1 8 0 3 4 3 7 8 1 3 4 1 6 8 8 3 4 7 1 3 7 1 0
5 7 4 0 6 8 5 5 2 6 7 4 3 5 0 7 2 3 7 6 7 2 1 3 7 3 5 6 1 4
4 6 0 7 5 8 1 1 3 0 2 2 5 5 6 0 7 0 6 0 0 1 7 2 7 8 2 0 6 3
5 3 2 3 5 8 5 5 4 8 0 2 5 8 6 6 1 5 8 0 7 5 2 3 1 0 1 3 4 2
7 2 4 6 7 4 1 3 5 5 8 6 3 6 7 8 8 6 2 4 8 4 2 1 7 3 4 6 4 0
0 7 7 4 5 6 5 2 5 6 4 0 0 2 1 1 7 8 1 6 6 4 7 5 0 2 8 8 2 5
1 3 3 7 8 8 4 8 0 4 7 4 8 1 0 8 5 7 8 7 5 0 4 6 6 2 7 6 1 2
```

OUTPUT

```
325
332
338
329
339
328
342
317
350
```

Sample Code Answer

```c
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

#define FLOORS 100
#define ROOMS 30

int main()
```

```
{
    int people[9] = {0,0,0,0,0,0,0,0,0};

    for (int i = 0; i < FLOORS; i++)
    {
        for (int j = 0; j < ROOMS; j++)
        {
            int input;
            scanf("%d", &input);
            people[input]++;
        }
    }
    for (int i = 0; i <= 8; i++)
    {
        printf("%d\n",people[i]);
    }

    printf("\n");
    return 0;
}
```

Q2 . The Grand Budapest Hotel has a 100 floors. Each floor has 30 rooms, and in each room there are 0 to 8 people. Given the number the number of residents in each room of the hotel, your task is to find the floor with the maximum number of empty rooms.

Input Format

There will be 100 lines, each line having 30 elements and each element lying between 0 and 8. The first line will represent the ground floor, the second line will represent the first floor and so on. You can think of it as a matrix `A` of size `100 x 30`. Where `a_i_j` represents the number of people in room j of floor i.

```
a_0_0  ...  a_0_29
a_1_0  ...  a_1_29
.
.
.
a_99_0 ...  a_99_29
```

Output Format

Once you find the floor with maximum number of empty rooms, print the floor number to output. If there are many floors with same maximum print the smallest floor number. If there are no floors with empty rooms print -1 to output.

```
i
```

Constraints

```
0 <= i < 100
0 <= j < 30
0 <= a_ij <= 8
```

Sample Test Cases

1. Sample Test Case 1

INPUT ->

```
0 2 4 4 4 6 5 4 1 3 4 6 3 7 6 3 1 2 5 3 0 2 2 7 1 7 6 0 1 0
2 6 4 7 7 0 2 6 7 4 7 1 1 1 3 6 0 5 0 6 2 4 5 1 1 6 4 0 1 3
6 5 5 7 5 6 5 4 0 6 1 0 0 5 6 7 2 1 5 5 2 5 5 1 7 3 5 6 7 5
3 1 4 1 3 7 6 3 4 5 6 3 5 0 0 3 1 6 7 5 3 0 5 5 3 6 1 1 0 4
0 6 2 5 1 7 4 1 2 0 3 1 4 5 4 5 2 4 5 3 3 2 7 0 1 2 0 2 7 7
2 4 6 4 3 7 5 0 1 3 6 3 3 7 4 0 1 4 0 6 7 7 3 4 7 7 5 0 7 2
3 2 2 0 1 2 2 5 0 0 2 7 0 4 3 7 7 7 4 0 1 6 2 6 1 5 1 2 2 5
4 7 4 6 0 4 2 2 3 7 0 1 1 0 3 4 6 2 4 3 2 1 5 0 4 3 7 0 6 1
7 1 0 1 0 3 1 5 0 5 0 7 4 3 0 1 7 3 0 5 7 7 0 3 7 0 7 1 0 0
4 7 2 6 6 5 1 0 5 5 1 0 2 6 1 4 2 3 7 4 5 4 0 2 0 3 0 0 0 7
2 1 7 2 4 1 2 0 6 4 3 3 4 7 5 4 7 7 0 6 6 6 1 4 0 5 5 2 1 0
6 2 4 5 7 4 0 6 7 4 7 4 7 2 1 2 7 3 5 0 4 6 6 2 4 4 4 1 3 7
3 7 1 0 4 4 5 6 1 3 2 1 1 0 2 3 3 2 4 1 1 5 3 5 7 7 0 4 2 7
4 3 7 0 7 7 5 7 4 7 7 7 1 5 5 2 2 5 0 4 5 3 4 4 6 2 7 3 0 1
5 4 5 7 3 3 2 0 2 7 7 1 3 2 5 4 0 3 7 6 5 0 6 2 4 6 7 6 4 0
6 3 7 0 6 7 5 7 7 3 6 3 2 3 7 2 3 1 5 6 3 3 4 0 3 5 5 3 2 1
3 0 7 4 1 6 4 1 2 1 4 1 4 6 5 7 5 4 2 1 6 3 2 2 4 6 7 3 6 7
0 7 1 3 6 6 5 2 6 2 4 3 1 7 4 2 1 5 7 6 4 4 2 1 7 2 1 3 5 2
7 2 1 6 6 7 4 0 1 6 1 1 5 1 3 1 2 6 4 2 7 7 2 5 1 2 4 4 2 5
2 4 7 7 2 1 1 5 0 4 6 1 5 5 1 2 0 5 5 7 3 7 4 5 4 6 1 3 4 7
4 3 0 1 6 7 4 0 2 2 5 3 7 4 6 4 0 6 4 5 3 3 7 7 7 1 7 1 4 3
4 7 3 0 1 1 2 7 7 1 0 2 6 7 1 6 4 0 7 7 7 6 0 0 4 4 4 5 0 5
6 1 0 2 7 2 5 6 5 3 4 1 0 4 5 4 5 6 6 2 6 4 4 0 0 7 6 1 5 5
5 2 2 2 0 2 2 1 5 7 0 1 0 2 0 3 0 1 0 4 2 6 4 2 2 1 6 5 4 5
5 0 5 7 2 0 4 4 3 2 4 3 1 0 0 1 5 1 4 5 2 0 6 4 2 7 5 4 4 7
1 3 1 0 3 6 3 4 4 0 0 4 7 7 0 5 3 5 3 1 6 0 0 2 4 2 4 1 2 0
7 0 7 2 4 1 7 1 2 2 4 0 4 6 6 2 4 3 4 0 3 5 2 0 1 1 6 6 7 7
2 0 0 0 2 0 7 4 5 3 2 7 4 7 3 1 7 4 5 5 0 7 7 2 2 2 6 2 2 4
6 3 6 3 0 3 4 2 2 5 3 0 2 4 1 7 1 0 0 2 2 2 4 0 3 6 0 7 6 3
3 3 6 0 5 7 6 0 7 6 0 4 6 0 2 0 0 6 2 3 7 5 5 1 6 5 7 1 5 3
1 3 0 1 6 4 4 0 6 4 6 1 7 3 5 3 0 3 2 7 4 7 6 2 1 4 4 2 0 2
3 5 4 1 1 3 6 2 3 7 6 5 5 3 5 7 7 2 7 0 4 6 7 7 1 4 0 3 4 1
4 3 0 7 2 7 6 0 6 5 6 4 2 2 0 2 5 0 4 7 4 1 5 7 3 6 4 6 1 4
7 5 7 4 4 4 5 0 2 0 1 6 0 7 5 3 3 2 0 6 2 0 0 7 0 4 1 5 7 4
1 3 0 1 0 7 5 7 1 2 5 7 2 5 6 1 4 1 2 4 6 5 5 5 7 4 7 5 3 7
5 5 2 4 0 4 7 1 1 2 4 6 5 7 7 3 3 7 4 7 1 2 3 0 1 3 1 6 4 6
5 3 6 0 7 4 4 7 0 7 5 5 6 7 6 6 6 5 4 7 2 0 5 5 3 3 1 2 0 3
6 0 3 2 4 1 3 6 0 0 0 2 2 5 3 4 6 3 3 7 0 1 6 5 1 4 0 4 0 6
2 1 7 4 7 5 7 0 3 7 6 6 2 0 4 5 4 1 4 5 4 6 1 7 3 0 4 2 2 7
3 0 1 6 7 1 4 1 6 6 2 4 1 5 6 0 5 2 6 1 4 5 3 6 5 4 5 3 6 4
1 1 3 4 0 3 7 4 6 6 7 5 1 7 5 4 5 0 5 7 4 2 7 6 1 5 0 6 2 3
0 6 7 5 5 1 3 2 0 4 6 0 5 5 3 4 4 5 6 7 1 4 2 1 2 6 5 3 4 4
7 5 2 1 3 3 0 1 3 7 1 6 5 6 7 7 3 4 4 3 4 6 6 2 6 6 0 0 0 5
3 0 3 3 7 1 0 5 7 5 2 0 3 4 7 4 1 1 7 1 3 0 7 3 3 4 0 1 3 1
3 7 3 0 0 5 4 2 3 3 2 7 4 0 3 1 0 7 0 5 6 6 2 5 0 7 7 5 4 3
2 4 1 5 6 0 4 7 5 2 7 4 0 3 1 5 3 6 4 7 0 5 3 5 7 5 1 5 1 7
3 0 4 0 7 4 5 4 7 0 1 0 2 2 2 5 5 7 4 5 0 2 7 0 0 4 1 6 5 5
```

6 4 6 6 7 1 4 4 4 1 6 7 5 2 1 2 0 3 4 1 2 5 1 0 3 0 1 6 0 0
5 7 6 3 6 5 0 4 0 5 7 7 0 4 1 2 2 7 1 5 2 3 1 5 0 4 1 5 7 2
1 5 4 2 4 7 6 4 2 1 3 4 5 4 0 2 2 7 7 3 7 5 4 1 2 0 7 3 1 6
6 0 5 3 3 5 1 3 1 6 2 5 1 4 0 3 2 7 1 3 7 5 3 5 3 2 0 4 2 3
7 3 4 7 3 1 2 0 2 7 6 0 3 7 5 7 1 0 3 2 2 7 2 0 4 4 1 6 5 2
4 2 7 1 4 5 2 0 0 2 6 0 5 4 6 6 3 7 3 6 2 6 6 4 1 0 3 6 2 1
3 1 4 6 5 0 7 1 1 1 0 3 6 2 5 6 4 5 0 6 2 3 2 0 2 5 6 7 7 5
7 5 2 7 7 3 7 3 5 6 3 0 7 1 7 6 0 6 6 6 4 4 3 2 6 4 4 5 6 3
5 7 2 6 1 4 3 4 0 2 6 6 1 1 5 6 1 6 6 3 7 6 0 6 5 3 7 1 5 6
2 2 4 2 7 7 3 3 6 6 2 5 1 7 7 0 7 4 4 2 3 4 5 3 4 6 3 1 5 2
0 7 7 2 6 1 5 7 0 5 7 2 0 7 2 1 6 6 3 4 6 4 2 5 2 5 7 4 5 3
1 5 6 6 6 7 3 5 6 1 3 1 4 6 3 7 7 0 4 5 2 1 2 6 3 0 0 0 1 0
2 3 0 4 0 1 1 1 5 5 0 0 2 6 0 0 4 2 6 7 0 5 3 1 1 3 2 1 4 1
0 6 5 6 5 6 5 0 5 2 3 5 0 5 6 7 2 5 3 3 6 1 6 2 1 5 2 3 2 1
7 1 7 7 5 6 3 1 3 5 1 2 3 6 3 7 6 7 1 7 7 6 5 5 1 0 2 4 5 0
4 3 2 7 4 7 2 2 5 6 0 6 0 2 6 5 5 1 6 6 6 2 0 7 6 0 5 5 0 5
2 2 3 6 4 1 4 0 0 0 1 4 3 5 4 7 6 1 0 5 2 3 2 6 3 7 3 0 7 0
4 1 3 7 6 0 1 1 5 7 4 0 7 2 2 4 6 5 1 0 4 4 5 5 5 3 5 0 1 2
3 2 0 4 5 7 6 1 1 2 4 2 2 4 7 0 3 5 3 4 0 7 2 5 0 1 5 1 5 0
5 0 6 0 2 4 2 4 6 7 6 6 6 3 6 7 1 2 7 3 6 5 4 7 1 0 5 0 2 1
7 6 7 1 2 7 0 7 1 1 6 7 2 6 4 4 1 0 4 1 0 7 1 6 6 2 0 5 0 4
5 3 3 0 1 1 1 7 6 1 1 6 5 5 5 2 6 6 7 2 3 4 6 5 0 1 5 0 6 3
0 6 2 1 5 3 2 0 2 6 5 6 0 7 2 2 4 6 3 3 7 5 7 0 4 1 5 2 4 2
0 5 1 7 0 7 0 2 3 5 1 4 2 0 3 5 3 5 7 5 7 0 7 7 0 3 0 0 0 5
0 1 2 0 5 0 1 6 3 7 1 0 4 6 2 6 6 3 3 5 1 2 5 1 6 5 2 3 6 6
5 0 4 7 2 2 1 7 6 4 7 5 4 7 7 7 4 5 3 5 3 0 4 6 6 1 2 7 4 1
7 6 5 7 1 0 6 5 5 6 5 7 7 0 3 3 4 4 1 3 1 2 6 7 2 4 3 6 2 4
7 1 2 5 3 6 3 5 4 4 4 6 1 1 2 5 4 4 2 6 3 6 2 6 3 3 0 2 0 6
3 3 7 0 7 1 0 1 1 6 2 7 0 3 2 1 3 3 3 6 5 6 5 2 0 0 2 1 4 2
2 2 5 1 3 0 3 6 1 2 1 3 6 0 4 7 0 4 0 0 4 3 3 3 0 1 7 3 0 1
4 7 4 6 0 5 6 0 7 3 4 4 1 7 7 3 7 6 4 4 7 3 6 5 6 0 4 4 0 5
2 7 0 0 1 2 2 6 5 1 4 0 5 7 1 7 3 7 7 5 4 7 1 7 4 0 2 3 2 5
3 2 3 3 5 4 6 4 6 3 1 3 4 0 5 1 1 1 0 0 3 7 6 2 0 3 0 0 7 0
0 7 1 4 5 1 3 0 2 2 2 7 6 5 1 2 4 4 4 5 3 2 0 2 7 0 6 6 2 0
2 4 7 4 0 6 5 4 3 4 1 7 7 2 6 5 2 6 2 2 4 2 1 5 2 1 0 5 5 4
4 3 6 2 0 3 0 0 1 0 7 5 0 1 0 1 2 6 5 4 0 2 2 4 1 3 7 1 6 1
5 2 2 0 5 5 4 3 4 1 5 5 5 4 4 6 3 1 3 0 2 1 7 0 2 3 4 1 3 7
1 3 0 2 2 2 4 3 0 4 6 7 7 2 6 4 7 7 0 7 0 0 4 4 6 2 4 0 4 2
6 3 4 7 7 1 7 0 2 6 6 7 2 5 3 1 5 1 3 1 6 5 2 0 2 4 4 1 7 6
5 0 1 7 1 6 2 4 7 0 6 0 0 6 0 0 2 4 6 7 2 0 4 7 6 3 3 2 3 5
6 6 4 3 1 4 1 4 6 7 2 3 3 7 7 2 3 2 5 3 5 4 4 6 7 1 7 3 5 6
7 2 4 7 0 1 0 2 4 4 5 2 7 1 2 0 6 5 0 1 7 1 0 5 2 3 1 7 5 7
7 5 5 7 6 6 2 4 4 2 7 5 6 2 0 3 2 5 3 2 2 4 4 4 7 3 7 5 4 6
7 3 4 6 6 1 7 0 4 4 2 4 4 4 0 4 5 2 6 5 5 2 0 7 4 6 0 2 1 0
4 6 2 6 4 1 3 0 0 0 0 7 0 0 4 7 1 2 6 5 6 2 5 1 0 1 3 3 2 0
5 7 0 6 2 4 6 6 4 7 6 2 6 6 0 0 3 7 1 1 0 7 2 1 3 5 0 7 0 2
6 3 5 7 4 6 6 5 0 7 3 6 0 1 3 2 7 6 6 1 2 6 4 0 4 1 0 0 0 7
1 4 5 7 2 6 0 7 0 3 1 6 4 6 2 4 4 4 4 4 4 0 2 4 0 6 5 3 4 2
3 7 6 6 2 0 6 2 5 5 4 3 4 2 7 7 7 5 4 2 1 0 3 4 4 2 6 6 4 3
6 0 5 7 7 5 7 5 2 4 3 1 0 7 2 0 7 5 0 0 5 0 5 0 1 2 2 0 7 4
2 3 1 5 2 1 1 3 3 5 0 3 4 3 2 6 2 7 6 5 3 5 7 3 5 6 7 3 1 4

```
3 3 4 4 2 3 0 0 2 0 1 0 3 4 4 6 0 7 2 4 4 7 3 4 2 0 2 1 3 1
3 5 1 4 4 3 0 3 6 2 3 1 2 4 3 1 4 3 5 0 6 0 0 5 2 1 3 6 3 1
```

OUTPUT

```
8
```

2. Sample Test Case 2

INPUT ->

```
4 7 6 4 5 3 3 4 1 1 4 2 4 5 3 1 7 1 1 2 2 4 3 1 3 5 6 5 5 5
6 5 7 1 4 5 5 1 7 3 1 7 1 7 3 5 1 5 1 3 4 7 2 2 6 6 2 6 3 6
6 4 5 1 3 3 3 3 6 6 5 3 1 4 1 2 4 7 3 7 7 5 4 2 5 2 1 5 6 5
6 4 4 5 3 4 6 3 4 6 5 2 1 2 3 2 5 7 6 2 6 4 3 3 2 6 1 2 4 3
5 6 7 5 1 2 2 4 2 2 2 2 5 7 7 2 5 5 4 5 5 4 1 7 7 6 2 5 4 2
7 3 1 7 7 7 3 3 3 6 1 1 7 6 4 3 6 2 3 6 4 1 5 6 7 1 3 6 7 3
1 5 2 6 3 2 5 5 7 2 1 5 3 5 1 2 1 4 4 2 1 1 1 3 3 3 6 6 5 4
4 6 4 2 6 1 5 3 3 7 4 1 5 4 5 2 3 6 6 1 6 4 5 1 1 1 3 3 5 6
7 4 7 6 3 3 3 6 6 3 4 3 2 2 7 4 3 4 2 3 1 3 5 5 6 3 1 5 2 1
2 6 1 3 4 3 4 7 6 1 1 3 2 6 1 3 6 6 2 2 3 7 2 2 6 2 2 5 3 1
6 2 7 6 3 3 2 6 7 2 6 7 7 4 7 4 3 7 2 3 4 3 4 5 1 2 7 1 4 4
4 3 4 2 6 3 5 6 5 4 6 7 2 7 2 1 1 3 6 3 7 1 7 5 1 6 7 7 2 1
6 7 7 3 7 5 2 5 3 7 3 6 5 7 7 6 3 2 6 3 6 2 2 6 5 6 4 2 4 5
4 3 7 3 4 3 6 4 1 3 3 5 7 6 3 2 1 2 5 6 2 2 5 7 7 1 5 4 4 6
2 3 5 5 1 7 6 4 3 3 2 3 4 4 5 5 5 7 2 5 3 4 3 6 2 3 5 5 4 5
6 2 7 3 1 2 4 7 4 4 4 5 4 7 3 7 2 6 5 5 2 4 5 3 1 6 4 2 2 6
3 4 1 5 3 2 1 3 6 6 1 1 6 2 4 7 4 6 6 7 6 1 5 6 2 7 3 2 4 2
7 7 2 5 4 7 7 5 6 1 3 6 5 1 2 3 1 4 5 3 4 3 5 4 4 3 6 7 4 4
7 2 4 6 4 1 1 6 1 7 3 1 7 7 2 4 7 5 7 3 7 4 6 3 7 7 6 3 1 4
5 7 4 3 4 4 2 6 5 6 3 7 1 2 1 5 4 1 7 2 5 2 3 7 7 5 4 7 2 5
1 4 1 1 1 2 1 1 3 2 2 5 6 2 7 3 5 6 6 5 5 2 5 5 5 2 2 6 5 1
6 5 3 5 3 3 6 5 7 3 2 5 1 5 2 2 5 6 3 1 5 5 7 2 3 2 1 1 5 6
7 6 3 6 5 6 6 5 3 2 3 4 5 7 6 3 2 1 5 5 3 7 2 5 2 4 7 5 3 2
4 1 1 2 1 2 2 2 4 3 7 2 7 5 6 6 1 4 6 6 5 3 6 6 4 7 4 7 6 6
6 2 7 2 5 2 6 4 3 1 2 6 3 2 1 3 6 1 3 5 5 7 7 1 5 7 2 3 4 4
3 7 2 4 2 1 5 2 6 7 3 6 6 7 6 1 3 1 3 5 2 6 1 5 1 1 3 2 1 1
1 3 2 3 2 3 5 2 6 5 1 5 4 4 6 4 1 2 1 7 7 2 2 7 6 3 1 2 5 7
5 4 6 4 6 4 6 3 7 6 6 6 2 4 6 2 1 1 6 5 6 4 1 2 2 7 6 5 7 6
6 1 7 4 3 1 6 5 1 2 3 5 1 7 2 6 4 5 2 5 7 6 4 6 3 7 7 1 2 1
2 2 1 7 5 7 5 4 3 2 2 2 1 5 6 6 5 3 3 6 2 4 6 6 5 2 1 4 5 6
3 3 1 1 1 7 4 2 2 2 5 5 3 5 1 6 4 3 1 4 5 1 3 7 7 3 2 2 2 5
5 2 5 4 7 3 3 3 1 7 7 4 2 3 6 7 3 2 7 1 6 4 4 4 4 6 1 7 2 7
6 2 3 7 6 2 1 6 3 6 5 7 6 6 3 4 1 1 6 7 2 1 4 5 2 7 5 7 1 4
6 1 4 2 5 7 3 2 1 7 5 6 2 5 2 7 3 3 2 3 4 1 5 5 7 7 3 3 3 3
2 1 1 2 2 1 7 7 7 1 1 3 4 7 4 1 6 7 1 6 4 6 6 3 7 2 6 5 6 7
6 7 1 5 5 6 5 4 6 3 6 2 5 6 4 5 6 7 5 7 3 1 4 2 1 5 5 4 1 3
3 1 1 3 5 4 5 7 6 4 4 6 4 1 3 4 5 1 3 5 7 7 3 7 2 5 1 1 2 5
4 6 3 5 7 6 3 4 3 2 1 1 5 5 6 7 4 1 5 2 6 4 1 3 4 2 7 1 7 5
5 5 3 1 7 7 7 1 5 1 1 3 7 6 2 7 1 3 6 2 4 2 4 1 4 4 6 5 7 7
5 6 4 5 4 4 2 3 4 7 3 3 6 7 1 4 1 7 6 5 5 6 5 2 2 6 1 5 4 2
5 4 6 7 1 4 1 1 6 3 5 5 3 5 2 6 5 4 4 3 3 6 1 1 2 3 4 5 7 5
2 7 2 4 2 4 7 7 1 5 5 1 1 1 4 2 1 6 3 2 5 1 7 3 3 4 6 7 6 3
```

3 3 5 2 5 3 5 1 7 3 1 4 4 1 1 5 4 5 4 2 4 7 1 5 1 7 6 5 1 4
6 3 1 5 2 1 3 6 6 1 3 4 2 6 6 3 6 3 7 1 6 7 7 5 4 1 6 2 1 3
3 6 2 5 6 5 3 7 3 2 1 4 2 3 7 6 7 1 4 6 4 7 5 1 2 2 1 7 2 6
7 2 3 6 5 4 7 5 7 2 3 5 2 3 3 4 6 2 2 6 7 1 7 2 6 4 6 5 3 7
3 4 3 6 1 3 1 3 2 4 7 7 5 6 2 1 3 2 7 3 2 7 5 2 1 6 3 2 1 5
7 4 2 6 6 4 4 5 6 3 4 1 4 7 4 6 5 2 3 3 7 3 1 5 4 5 5 7 1 6
3 2 3 2 5 5 4 4 3 3 3 5 5 5 2 5 2 1 5 7 4 4 4 7 2 1 7 2 2 7
1 3 5 4 6 7 4 4 7 1 3 6 2 1 1 4 2 6 3 4 7 4 3 7 1 2 5 1 5 4
2 4 7 3 1 5 2 5 3 5 4 1 2 1 2 5 6 1 4 6 1 3 3 3 3 6 2 4 5 3
1 5 5 7 3 1 4 7 6 6 3 5 6 3 7 2 5 3 7 2 3 2 3 3 5 3 2 6 1 3
7 6 5 1 1 4 4 4 7 3 2 5 3 6 1 5 2 5 5 6 7 5 1 3 3 5 5 3 3 6
1 1 2 3 7 3 2 7 4 3 7 3 2 1 4 7 5 7 4 7 6 7 2 3 3 3 3 4 6 5
6 7 1 5 5 3 4 3 2 6 4 5 3 6 3 5 3 1 4 4 7 2 2 2 2 6 3 4 4 6
4 1 6 3 6 1 2 7 3 6 4 5 4 2 3 2 2 7 7 1 2 5 6 1 3 3 4 3 5 1
6 1 1 6 2 7 6 1 5 1 3 5 3 1 1 2 4 3 3 3 7 3 6 2 7 5 5 1 4 6
1 5 3 7 5 4 1 4 4 2 1 1 4 1 6 1 7 1 7 4 2 1 6 3 5 2 1 6 2 3
1 1 5 7 6 6 7 1 5 1 7 6 3 6 3 4 7 7 1 5 1 6 7 4 7 6 1 6 5 3
6 4 7 5 7 4 4 1 4 2 5 3 5 4 7 1 5 7 4 1 5 1 3 3 4 5 6 1 1 6
6 5 7 3 1 3 4 4 4 4 5 3 6 2 2 1 5 4 4 6 6 1 5 3 6 3 6 5 3 5
7 3 2 7 4 2 7 4 6 3 3 7 4 5 1 4 5 1 5 4 6 2 7 1 5 5 2 3 7 4
1 2 6 6 2 7 1 5 6 4 4 1 1 1 1 4 5 5 6 4 1 1 5 4 4 5 7 5 1 4
1 2 2 1 4 7 1 1 1 2 3 6 2 7 7 3 1 2 7 4 4 5 1 6 5 3 6 5 3 6
6 4 2 4 6 1 3 1 1 7 3 3 2 2 7 7 1 3 6 6 4 5 6 4 2 2 7 5 1 1
7 5 7 1 1 6 2 7 3 2 7 7 4 1 2 1 5 2 4 1 6 6 4 1 7 3 5 7 3 4
2 6 4 3 3 6 5 7 2 4 1 5 2 1 2 6 4 6 6 2 6 2 6 7 5 1 7 5 5 5
6 5 3 2 1 4 1 4 2 1 1 4 7 6 6 6 7 4 4 7 6 1 2 5 7 4 2 7 7 1
6 7 7 2 4 1 6 2 3 2 1 6 2 6 5 7 3 7 2 6 3 7 1 1 6 3 2 3 7 7
7 1 5 1 6 2 5 2 2 6 4 1 1 1 7 2 3 7 7 2 5 6 3 4 7 2 2 5 1 4
5 2 2 3 2 1 4 1 3 6 7 7 3 4 2 1 1 5 1 4 3 1 7 7 6 3 4 7 4 7
3 2 7 2 5 6 2 4 7 1 1 4 6 3 6 2 7 7 7 4 5 5 3 2 1 7 3 2 5 5
7 1 4 1 4 5 3 7 7 2 5 4 4 2 6 1 7 7 7 1 5 4 5 5 2 2 4 7 4 6
5 6 7 7 3 3 2 5 3 6 2 1 4 7 4 3 3 6 7 7 6 7 1 6 3 7 3 7 1 6
7 1 2 4 6 7 4 1 7 7 2 6 3 1 6 3 6 4 3 1 5 7 3 7 1 6 5 5 6 4
3 1 6 1 5 5 7 6 2 7 1 3 7 1 5 6 3 2 6 5 2 2 5 7 2 4 6 4 7 4
1 6 1 3 7 4 6 2 2 1 7 5 2 1 1 2 4 2 1 6 3 2 4 1 6 3 7 2 4 6
6 7 3 3 5 3 7 7 7 5 2 6 5 5 1 4 2 7 2 3 1 7 3 7 5 1 3 4 2 4
1 3 1 1 6 4 5 6 3 3 3 5 6 1 3 1 1 2 7 2 1 7 2 4 4 2 5 5 3 6
5 3 2 1 6 7 7 7 7 7 5 7 5 7 4 2 6 5 5 5 1 2 7 4 4 2 3 1 3 3
5 7 1 6 6 1 4 4 1 4 1 2 1 7 1 2 6 6 6 4 5 1 3 2 5 6 3 2 6 4
4 3 6 5 7 6 3 1 2 4 1 1 3 1 4 3 6 3 4 4 6 4 7 2 6 6 3 4 7 4
4 2 1 4 4 6 2 7 2 3 1 7 4 2 5 6 3 7 6 1 7 5 2 4 2 4 5 2 6 2
6 1 2 2 7 7 1 6 1 6 6 7 3 6 3 2 4 6 6 1 2 3 1 6 1 5 3 3 5 1
5 3 7 1 3 6 5 6 5 7 6 4 5 7 6 2 4 3 1 3 4 1 1 5 2 7 3 2 4 3
4 3 5 1 2 7 1 3 4 4 6 5 4 5 1 1 7 2 1 4 3 5 5 7 7 4 5 4 5 1
3 2 3 1 4 7 3 3 6 5 7 5 3 6 7 4 4 4 1 4 6 2 3 2 1 3 3 1 5 4
1 5 3 4 5 7 4 4 1 1 1 1 6 6 3 4 7 6 7 3 3 3 6 1 3 4 6 7 7 7
4 2 5 4 3 2 1 6 2 2 1 3 3 3 1 7 2 3 2 7 1 5 3 1 7 5 4 1 7 4
7 5 5 4 7 7 2 4 7 2 5 7 6 6 7 6 5 5 2 1 1 7 6 2 5 1 3 3 1 5
1 5 3 1 2 3 3 2 6 2 3 2 6 4 4 3 4 1 3 3 5 6 6 4 1 1 6 5 2 2
3 6 2 3 2 4 4 1 2 5 1 2 2 4 5 5 5 1 1 6 7 1 1 4 1 4 5 1 3 2
1 1 3 1 3 3 4 6 3 4 4 4 5 1 4 2 7 5 4 7 1 3 2 3 7 3 2 5 4 6
4 4 4 4 2 3 2 4 2 5 4 4 3 4 7 5 4 5 4 2 3 7 5 3 4 1 5 4 4 2

```
5 7 1 2 5 6 1 1 5 6 7 4 1 5 7 6 2 7 4 2 1 4 2 4 1 7 4 5 3 1
2 6 1 5 6 4 4 2 7 7 2 2 6 4 4 6 2 5 6 3 5 1 1 7 3 6 2 6 7 3
1 2 6 5 2 1 1 5 5 5 5 6 6 1 4 1 7 7 4 3 4 4 7 1 5 4 1 2 4 4
6 6 7 7 7 6 5 6 6 5 7 7 3 2 2 5 1 7 7 1 2 1 4 3 2 4 4 7 3 1
1 6 2 3 1 4 3 6 7 3 6 7 6 4 3 5 6 5 7 3 4 5 4 4 7 3 1 4 6 5
2 5 4 4 4 4 6 2 4 4 5 7 4 5 5 6 1 7 3 3 1 5 5 3 4 6 2 2 3 7
```

OUTPUT

```
-1
```

Sample Code Answer

```c
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

#define FLOORS 100
#define ROOMS 30

int main()
{
    int floor[100];
    int all_empty = 1;
    int max = FLOORS - 1;

    for (int i = 0; i < FLOORS; i++)
    {
        int empty = 0;
        for (int j = 0; j < ROOMS; j++)
        {
            int input;
            scanf("%d", &input);
            if (input == 0)
                empty++;
        }
        floor[i] = empty;
        if (empty)
            all_empty = 0;
    }
    if (all_empty == 1)
    {
        printf("-1");
        goto exit;
    }
    for (int i = FLOORS - 1; i >= 0; i--)
    {
        if (floor[max] <= floor[i])
            max = i;
    }
    printf("%d", max);

exit:
```

```
    printf("\n");
    return 0;
}
```