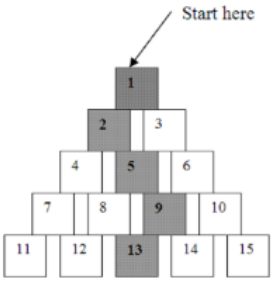
Pyramid Adventurer

A 2-Dimensional pyramid has a number of chambers as illustrated in the figure below. As an adventurer, you are required to work your way from the pinnacle of the pyramid to its bottom. Indicated inside a chamber is a number representing the reward that will be collected if you travel through that chamber on your way to the bottom. From each chamber, you are only allowed to move downwards to the chamber diagonally on the left or right. For example, from the chamber with reward 5, you can move to either chamber with reward 8 or 9. However, you cannot go to chamber with reward 7 or 10 from the one with reward 5. If you take the path as shown by the sequence of chambers shaded gray, the total reward will be 1+2+5+9+13 = 30.



Your task is to write a program that computes the number of paths contained within a given pyramid whose reward is equals a given number *R*.

Input Format

The input begins with a line containing two integers *N* and *R*. *N* represents the height of the pyramid, that is the number of rows in the pyramid and *R* represents the reward. The next N lines represents the chamber matrix which contains the reward values for each chambers in the row followed by a number of zeros to accomodate the rest of the chamber matrix.

Refer to the sample input for illustration.

Constraints

- **The height of the pyramid N**: A positive integer where (4 <= *N* <= 10).
- **The reward R**: A positive integer where (1 <= R <= 100).
- **The reward value of the chamber V**: A positive integer where (0 <= V <= 20).

Output Format

Output the number of paths whose reward equals to *R*.

Sample Input 0

```
5 29
1 0 0 0 0
2 3 0 0 0
4 5 6 0 0
7 8 9 10 0
11 12 13 14 15
```

Sample Output 0

```
2
```

Explanation 0

There are 2 paths in the pyramid whose reward equals to 29.