Pascal's triangle

Problem ID: a10p05pascalstriangle

Write a Python program that asks for the height of a Pascal's triangle then generates the triangle and prints it out. A Pascal triangle of height 6 looks like this:

```
                1
             1     1
          1     2     1
       1     3     3     1
    1     4     6     4     1
 1     5    10    10     5     1
```

Rules to generate Pascal's triangle:

1. The first row is always 1, and the second row is always 1 1.
2. After the first two rows, each row at level h is generated from the values at row $h - 1$.

The leftmost number and the rightmost number in any row are always 1. Note that, in row h, there are h numbers.

The main program is given, but you need to implement the function `make_new_row(row)`. It should take one list as an argument, one row of Pascal's triangle, and return a list, the next row in Pascal's triangle.

For example the expected output of a call to `make_new_row()` with `[1, 1]` as the parameter, is `[1, 2, 1]`.

Input

The input consists of one line containing one integer h, the height of the triangle, where $0 \leq h \leq 100$.

Output

The output consists of h lines, where the ith line contains the ith level of Pascal's Triangle. Each level should be output in Python list notation.

Sample Input 1

```
5
```

Sample Output 1

```
[1]
[1, 1]
[1, 2, 1]
[1, 3, 3, 1]
[1, 4, 6, 4, 1]
```

Sample Input 2

```
1
```

Sample Output 2

```
[1]
```

Sample Input 3

```
2
```

Sample Output 3

```
[1]
[1, 1]
```

Sample Input 4

```
10
```

Sample Output 4

```
[1]
[1, 1]
[1, 2, 1]
[1, 3, 3, 1]
[1, 4, 6, 4, 1]
[1, 5, 10, 10, 5, 1]
[1, 6, 15, 20, 15, 6, 1]
[1, 7, 21, 35, 35, 21, 7, 1]
[1, 8, 28, 56, 70, 56, 28, 8, 1]
[1, 9, 36, 84, 126, 126, 84, 36, 9, 1]
```

Sample Input 5

```
0
```

Sample Output 5