Easy problem:

Write a program in python to calculate the sum of the series (1*1) + (2*2) + (3*3) + (4*4) + (5*5) + ... + (n*n).

Sample Output:
Input the value for nth term: 5
5*5 = 25
4*4 = 16
3*3 = 9
2*2 = 4
1*1 = 1
The sum of the above series is: 55

Medium problem:
Write a program in python to make such a pattern like right angle triangle using number which will repeat the number for that row.

Sample Output:

Input number of rows: 5	**1**
22	**333**
4444	**55555**

Hint: use **print**('text', end = '') for printing without new line

Hard Problem:

Write a program in python to print a pyramid of digits as shown below for n number of lines.

Sample		**Output:**
Input the number of rows: 5		**1**
232	**34543**	**4567654**
567898765		

Challenging Problem:

In the magical land of Byteland, there are three kinds of citizens:

- a Bit - $2ms$ after a Bit appears, it grows up and becomes a Nibble (i.e. it disappears, and a Nibble appears)
- a Nibble - $3ms$ after a Nibble appears, it grows up and becomes a Byte
- a Byte - $5ms$ after a Byte appears, it grows up, splits into two Bits and disappears

We want to know the answer to the following question: what would the population of Byteland be immediately before the time Nms if only 1 Bit appeared at time $0ms\,0ms$?

Find the population (number of citizens) of each type from 0ms to Nms.

Input
- a single integer N.

Output
Print a single line containing four space-separated integers — time, the number of Bits, Nibbles and Bytes.

Example Input
```
5
```

Example Output
```
0 1 0 0

1 1 0 0

2 0 1 0

3 0 1 0

4 0 1 0

5 0 0 1
```