

## Logic Pyramid

Identify the logic behind the series

6 28 66 120 190 276....

The numbers in the series should be used to create a Pyramid.

The base of the Pyramid will be the widest and will start converging towards the top where there will only be one element. Each successive layer will have one number less than that on the layer below it. The width of the Pyramid is specified by an input parameter N. In other words there will be N numbers on the bottom layer of the pyramid.

The Pyramid construction rules are as follows:

First number in the series should be at the top of the Pyramid

Last N number of the series should be on the bottom-most layer of the Pyramid, with Nth number being the right-most number of this layer.

Numbers less than 5-digits must be padded with zeroes to maintain the sanctity of a Pyramid when printed. Have a look at the examples below to get a pictorial understanding of what this rule actually means.

Example

If input is 2, output will be

```
00006
00028 00066
```

If input is 3, output will be

```
00006
00028 00066
00120 00190 00276
```

Formal input and output specifications are stated below

Input Format:

First line of input will contain number k - the number of inputs.

The following k lines contain N that corresponds to the width of the bottom-most layer of each Pyramid

Output Format:

The Pyramid constructed out of numbers in the series as per stated construction rules

Constraints:

$0 < N \leq 14$

Code :

```
import java.util.*;
```

```
class Main{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        for(int i = 0 ; i < n ; i++){
            sc.nextLine();
            int k = sc.nextInt();
            int count = 1;

            for(int j = 1 ; j <= k ; j++){
                for(int m = 1 ; m <= ((k-j)*3);m++){
                    System.out.print(" ");
                }
                for(int l = 1; l <= j ; l++){
                    // System.out.print(" ");
                    System.out.print(String.format("%05d",element(count))+" ");
                    count++;
                }
                System.out.println();
            }
        }
    }
}
```

```
static int element(int count){
```

```
    int first = 6;
```

```
    int second = 28;
```

```
    if(count == 1) return first;
```

```
    if(count == 2) return second;
    int next= 0;
    for(int i = 3 ; i <= count ; i++){
        next = (second * 2) - first + 16;
        first = second;
        second = next;
    }
    return next;
}
```