Test 1

Number Star pattern 1

Send Feedback

Print the following pattern for given number of rows.

Input format :

Integer N (Total number of rows)

Output Format :

Pattern in N lines

Sample Input :

5

Sample Output :

5432*
543*1
54*21
5*321
*4321

```
import java.util.Scanner;
public class runner {
   public static void main(String[] args) {
       // Write your code here
       Scanner sc = new Scanner(System.in);
       int n = sc.nextInt();

       for (int i = 0; i < n; i++) {
           for (int j = n; j > 0; j--) {
               if (j == i + 1) {
                   System.out.print("*");
                   continue;
               }
               System.out.print(j);
           }
           System.out.println();
       }
   }
}
```

Zeros and Stars Pattern

Send Feedback

Print the following pattern

Pattern for N = 4

*000*000*
0*00*00*0
00*0*0*00
000***000

Input Format :

N (*Total* no. of rows)

Output Format :

Pattern in *N* lines

Sample Input 1 :

3

Sample Output 1 :

*00*00*

0*0*0*0

00***00

Sample Input 2 :

5

Sample Output 2 :

*0000*0000*

0*000*000*0

00*00*00*00

000*0*0*000

0000***0000

```
import java.util.Scanner;
public class Solution {
   public static void main(String[] args) {
       Scanner sc = new Scanner(System.in);
       int n = sc.nextInt();

       for(int i=0; i<n; i++){
           for(int j=0; j<n; j++){
               if(j==i){
                   System.out.print("*");
                   continue;
               }
               System.out.print("0");
           }

           System.out.print("*");

           for(int j=n-1; j>=0; j--){
               if(j==i){
                   System.out.print("*");
                   continue;
               }
               System.out.print("0");
           }

           System.out.println();
       }
   }
}
```

Check Armstrong

Send Feedback

Write a *Program* to determine if the given number is *Armstrong* number or not. Print true if number is armstrong, otherwise print false.

An Armstrong number is a number (with digits n) such that the sum of its digits raised to nth power is equal to the number itself.

For example,

371, as 3^3 + 7^3 + 1^3 = 371

1634, as 1^4 + 6^4 + 3^4 + 4^4 = 1634

Input Format :

Integer n

Output Format :

true or false

Sample Input 1 :

1

Sample Output 1 :

true

Sample Input 2 :

103

Sample Output 2 :

false

```
import java.util.Scanner;
import java.lang.Math;
public class Main {

   public static void main(String[] args) {
       // Write your code here

       Scanner sc= new Scanner(System.in);
       int num = sc.nextInt();
       int n = num, sum=0;
       int length = (int)Math.log10(num) + 1;

       while(n>0){
           sum += Math.pow(n%10,length) ;
           n /= 10;
       }
       if(sum == num)
           System.out.println("true");
       else
           System.out.println("false");
   }
}
```