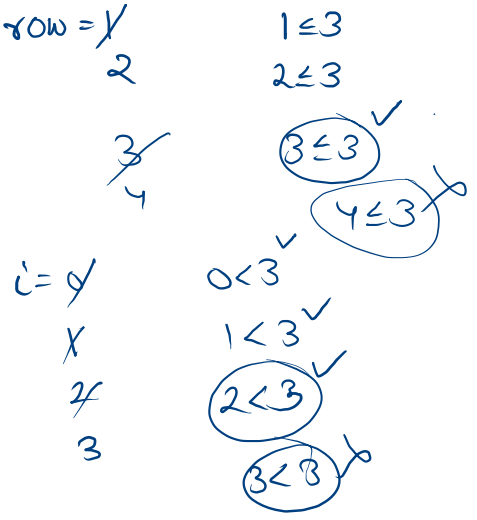
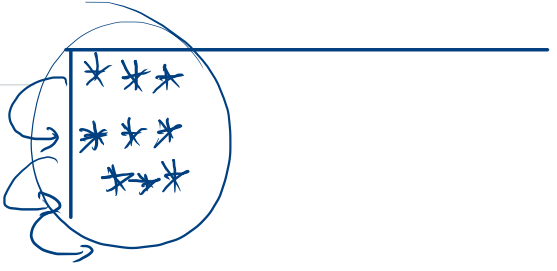
Pattern -3 ⇒ n * n

n=3


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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();

        for(int row = 1; row <= n; row++){
            //each row -- n stars
            for(int i = 0; i < n; i++){
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```





n = 2.

row = 1 2

1 <= 2 ✓

2 <= 2

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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();

        for(int row = 1; row <= n; row++){
            //each row -- n stars
            for(int i = 0; i < n; i++){
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

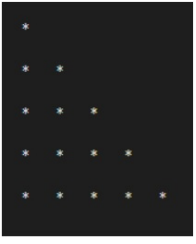
GKSTR19 Pattern_4

Sample Input 0

```
5
```

Sample Output 0

```
*
* *
* * *
* * * *
* * * * *
```



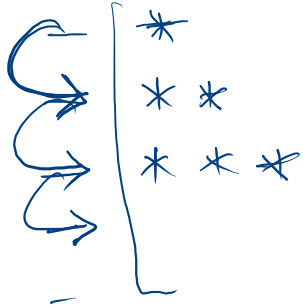
1. rows. → n
2. initial. → 1
3. update → star++

0 1 n n/2 - - -

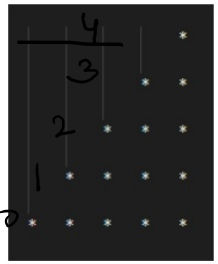
Star = X 2 3 4

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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();

        int star = 1;
        for(int row = 0; row < n; row++){
            for(int st = 0; st < star; st++){
                System.out.print("* ");
            }
            System.out.println();
            star++;
        }
    }
}
```

1 x n



Pattern - 5

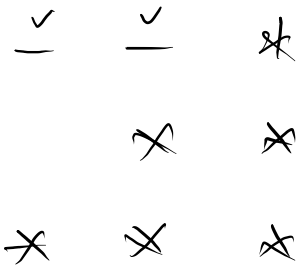


n = 5

rows → n

init → [star = 1
space = n-1

update → star ++
→ space --

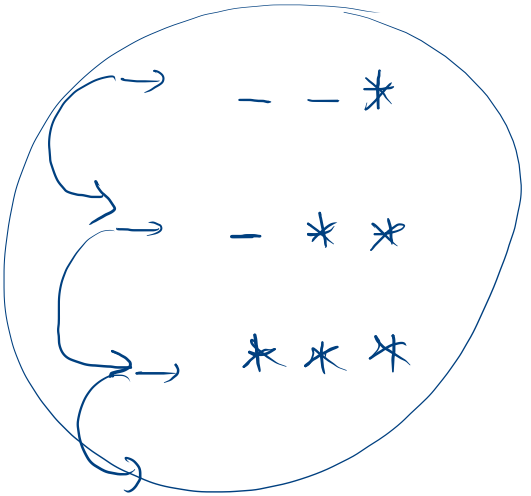


n = 3

star = 1 2 3 4

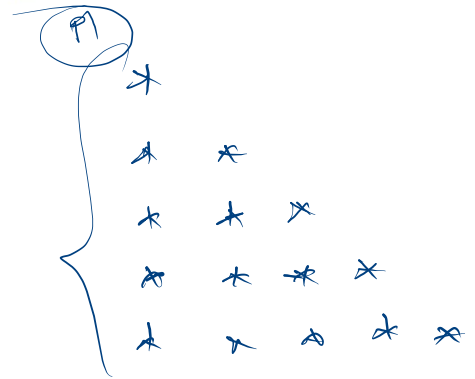
space = 2 1 0 -1

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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();

        int star = 1;
        int space = n-1;

        for(int row = 0; row < n; row++){

            for(int sp = 0; sp < space; sp++){
                System.out.print(" ");
            }


            for(int st = 0; st < star; st++){
                System.out.print("*");
            }
            System.out.println();
            star++;
            space--;
        }
    }
}
```



GKSTR17 Pattern_2

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```



n=2

```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         int n = scn.nextInt();
9
10        int star = 1;
11
12        for(int row = 0; row < n; row++){
13            int val = 1;
14
15            for(int st = 0; st < star; st++){
16                System.out.print(val + " ");
17                val++;
18            }
19            System.out.println();
20            star++;
21
22        }
23    }
24 }
```

n=3.

1 ✓
1 2
1 2 3

star = 1 2 3 4

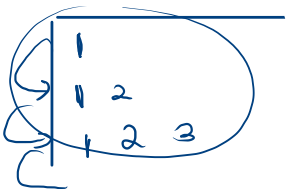
row = 0 1 2 3

2<3 ✓
3<3 ✗

val = 1 2 3 4

st = 0 1 2 3

0<3
1<3
2<3 ✓
3<3 ✗



Pattern 6 - Right triangle of 5 multiples

n= 5

5

5 10

5 10 15

5 10 15 20

5 10 15 20 25

n = 3

5

5 10

5 10 15

n = 2

5

5 10

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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();

        int star = 1;

        for(int row = 0; row < n; row++){
            int val = 5;

            for(int st = 0; st < star; st++){
                System.out.print(val + " ");
                val += 5;
            }
            System.out.println();
            star++;

        }
    }
}
```

GKSTR24 Pattern_7_Pyramid

n = 5

→ rows = n.

→ [star = 1
space = n-1

→ star ++
space - -

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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();

        int star = 1;
        int space = n-1;

        for(int row = 0; row < n; row++){
            for(int sp = 0; sp < space; sp++){
                System.out.print(" ");
            }

            for(int st = 0; st < star; st++){
                System.out.print("* ");
            }

            System.out.println();
            star++;
            space--;
        }
    }
}
```

Diamond.

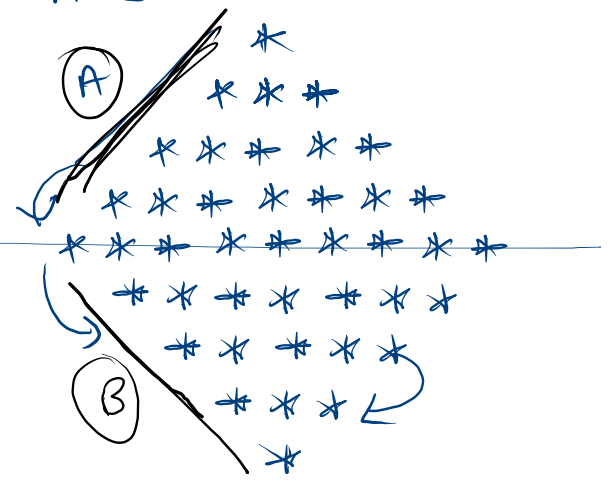
n → odd

2n-1

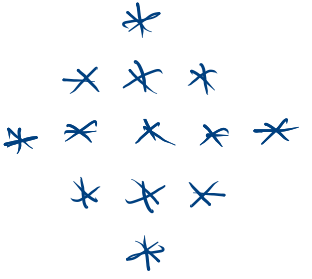
```
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

n= 5

n=5 → 9



n=3 → 5



rows → 2n-1

init

star = 1

space = n-1

Update.

star +2

space--

→ A

star -2

space++

→ B

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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int star = 1;
        int space = n-1;

        for(int row = 0; row < (2*n-1) ; row++){    //work for 2n-1 times
            for(int sp = 0; sp < space; sp++){
                System.out.print(" ");
            }
            for(int st = 0; st < star; st++){
                 System.out.print("*");
            }
            System.out.println();
            //update

            if(row < n-1){
                star += 2;
                space--;
            }else{
                star -= 2;
                space++;
            }

        }
    }
```