n = 3

i = 1

	1	2	3
1	3	2	1
2	3	2	
3	3		

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

n = 5

```
int n;

for(int i = 1; i <= n;i++){
    for(int j = n; j >= i; j--){
        Syso(j + " ");
    }
    sysoln();
}
```

Space = " "

tab space = "\t";

Inverted triangle

0 * * * * * * *

1 - * * * * *

2 - - * * *

3 - - - *

m = 7

0 row - 0

1 row - 1

2 row - 2

3 row - 3

Spaces

Stars Spaces ↑ (i)

Stars m = 7

k loop

2x 1 row - 5 7-2

2x 2 - 3 7-4

2x 3 - 1 7-6

(m - 2*i)

Hw_Print Inverted triangle

```java
Scanner s = new Scanner(System.in);
int m = s.nextInt();

for(int i = 0; i< m;i++){
    System.out.print("*" + "\t");
}
System.out.println();

for(int i = 1; i < m; i++){
    // spaces
    for(int j = 0; j < i; j++){
        System.out.print("\t"); // tab spaces
    }
    // stars with tab spaces
    for(int k = 0; k < m - 2 * i; k++){
        System.out.print("*" + "\t");
    }
    System.out.println();
}
```

m = 5

* * * * *

// first line

— outer

- inner 1

- inner 2

i = 1 < 5 T

j = 0 < 1 T

1 < 1 F

i = 2 < m T

j = 0 < 2 T

1 < 2 T

2 < 2 F

k = 0 < 3 T

1 < 3 T

2 < 3 T

3 < 3 F

k = 0 < 1 T

1 < 1 F

* * * * *

— * * *

— — *

i = 3 < 5 T

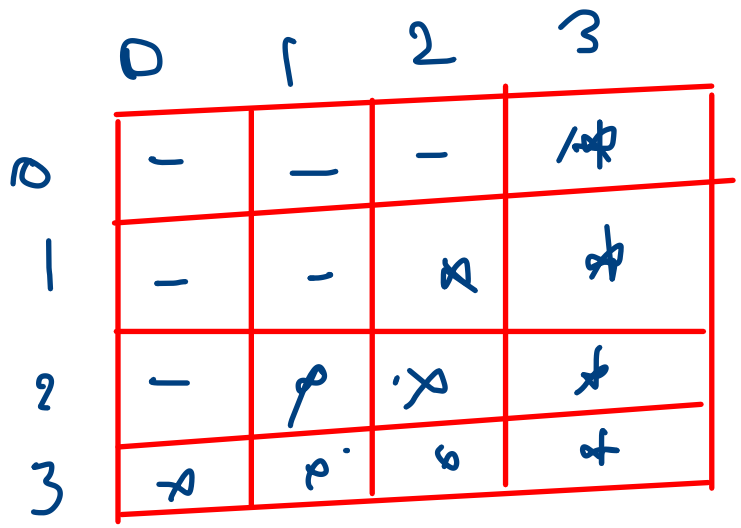
j = 0 < 3 T

1 < 3

2 < 3

k = 0 <

n=3



Diamond Pattern →

```
int m;
int lines = m/ 2 + 1;

// upper half including the middle line
for(int i = 0; i < lines; i++){
    for(int j = 0; j < i; j++){
        syso("\t");
    }
    for(int j = 0; j < m-2*i;j++){
        syso("*" + "\t");
        if(j < m-2*i - 1){
            syso("\t";)
        }
    }
    sysoln();
}

    // bottom half
    for(int i = lines -2; i >= 0;i--){
        for(int j = 0; j < i; j++){
            syso("\t");
        }
        for(int j = 0; j < m-2*i;j++){
            syso("*" + "\t");
            if(j < m-2*i - 1){
                syso("\t";)
            }
        }
        sysoln();
    }
```