



Today's agenda

↳ Patterns



AlgoPrep



Q) Print N "*" in a single row.

Ex: $N=4 \rightarrow ****$

N iteration \leftarrow

```
for (int i=1; i<=N; i++)  
    System.out.print("*");  
}
```



AlgoPrep



Q) Given Integer N , Print square of $N \times N$ using $"*"$.

Ex: $N=4$

* * * *

* * * *

* * * *

* * * *

$N=5$

* * * * *

* * * * *

* * * * *

* * * * *

* * * * *

$N=3$

```
for (int i = 1; i <= N; i++) {  
    for (int j = 1; j <= N; j++) {  
        System.out.print("*");  
    }  
    System.out.println();  
}
```

How many such rows.

What to print in 1 row

* * *

* * *

* * *

↳

i	i <= N	j	j <= N
1	+	1	+
		2	+
		3	+
		4	6 → exit
2	+	1	+
		2	+
		3	+
		4	6 → exit
3	+	1	+
		2	+
		3	+
		4	6 → exit
4	6		6 → exit



Q) Pattern 1:

↳ Print the triangle pattern.

$N=2$:

*
* *

$N=4$:

*
* *
* * *
* * * *

$N=3$

int nst = 1;

for (int i = 1; i <= N; i++) {

for (int j = 1; j <= nst; j++)
System.out.print("*");
}

Print for next line
}

nst++;
System.out.println();

*
* *
* * *

nst = 1 2 3 4

i	i <= N	j	j <= nst
1	t	1	t
		2	f
2	f	1	t
		2	t
		3	f
3	f	1	t
		2	t
		3	t
		4	f
4	f		



Q) Pattern 2

↳ Print the following Pattern.

$N=3$

```
  _ * x
    * * *
  _ * x
```

$N=5$

```
  _ _ _ * x x
+2↓  _ _ * * * x ↓-1
+2↓  _ _ * * * x ↓-1
+2↓ * * * * * x ↓-1
-2↓ _ _ _ * * * x ↓+1
-2↓ _ _ _ * x x x ↓+1
```

$N=7$

```
  +2↓ _ _ _ _ _ * x x x x x +2↓
+2↓ _ _ _ _ _ * * * x x x ↓-1
+2↓ _ _ _ _ _ * * * * * x ↓-1
+2↓ * * * * * * * * x ↓-1
-2↓ _ _ _ _ _ * * * * * x ↓+1
-2↓ _ _ _ _ _ * * * x x x ↓+1
-2↓ _ _ _ _ _ * x x x x x ↓+1
```

$N=6 \rightarrow$ incorrect input



$N=5$

```
— — * x x
— * * * x
* * * * *
— * * * x
— — * x x
```

```
int nst = 1;
int nsp = N/2;
```

```
for (int i = 1; i <= N; i++) {
```

```
    for (int j = 1; j <= nsp; j++) {
        System.out.print(" ");
    }
```

```
    for (int k = 1; k <= nst; k++) {
        System.out.print("*");
    }
```

```
    if (i <= N/2) {
        nst = nst + 2;
        nsp = nsp - 1;
    }
```

```
    else {
```

```
        nst = nst - 2;
        nsp = nsp + 1;
    }
```

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

nsf = x^2 x^{-1} nsP = $2x^2 0x^2 2x^3$

$$i < \frac{n}{2}$$

t



b

b

b**b**

6



Genit

Genit

Genit

Break till 10:43 PM



Q) Pattern 3

↳ Print the following Pattern.

$N=5$?

1 2 3 4 5 6 7
* * * _ * * *
* * _ _ _ * *
* _ _ _ _ *
* * _ _ _ * *
* * * _ * * *

$N=7$

1 2 3 4 5 6 7 8 9
1 * * * * _ * * * *
2 -1↓ * * * _ ↘+2 * * *
3 -1↓ * * _ ↘+2 * *
4 -1↓ * _ _ ↘+2 _ *
5 +1↓ * * _ _ ↘-2 * *
6 +1↓ * * * _ ↘-2 * * *
7 +1↓ * * * * _ ↘-2 * * * *

$$\frac{N+2}{2} = \frac{N}{2} + \frac{2}{2}$$

$$= \frac{N}{2} + 1$$



//Pseudo code

```
int nsp = 1;
```

```
int nst =  $\sqrt{n}/2 + 1$ ;
```

```
for (int i = 1; i <= n; i++) {
```

```
    for (int j = 1; j <= nst; j++) {  
        System.out.print("*");  
    }
```

```
    for (int j = 1; j <= nsp; j++) {  
        System.out.print(" ");  
    }
```

```
    for (int j = 1; j <= nst; j++) {  
        System.out.print("*");  
    }
```

```
    if (i <=  $\sqrt{n}/2$ ) {  
        nst = nst - 1;  
        nsp = nsp + 2;  
    }
```

```
    else {  
        nst = nst + 1;  
        nsp = nsp - 2;  
    }
```

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

```
}
```



Q) Pattern 4:

↳ Print the triangle Pattern with number

N=2:

1
2 3

N=4:

1
2 3
4 5 6
7 8 9 10

int count = 1;

int nst = 1;

for (int i = 1; i <= N; i++) {

for (int j = 1; j <= nst; j++)
System.out.print(count);
count++;
}

nst++;

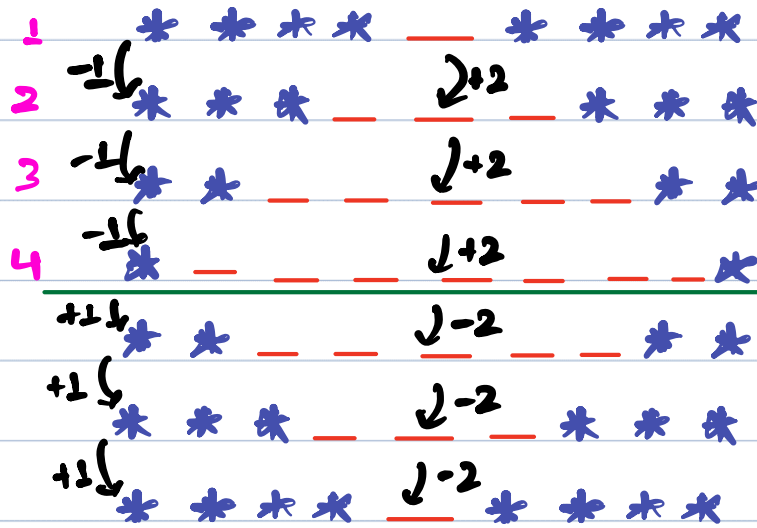
System.out.println();

}



$N=7$

1 2 3 4 5 6 7 8 9



$$\frac{N+2}{2}$$



AlgoPrep