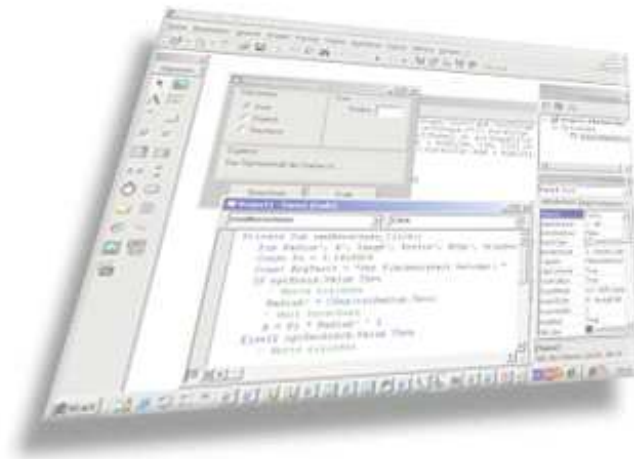


# Wiederholungen II: for-Wiederholungsanweisung





# Beispiel Ia

Ausgabe der Quadratzahlen 1 bis 100

```
int i;  
for(i=1;i<=10;i=i+1)  
    System.out.println(i*i);
```

1. Durchlauf

Setze i=1

Ausgabe von 1\*1

Da i noch nicht 11, springe zu der Zeile „for-...“



# Beispiel Ib

Ausgabe der Quadratzahlen 1 bis 100

```
int i;  
for(i=1;i<=10;i=i+1)  
    System.out.println(i*i);
```

2. Durchlauf

Setze i=2

Ausgabe von 2\*2

Da i noch nicht 11, springe zu der Zeile „for-...“



# Beispiel Ic

Ausgabe der Quadratzahlen 1 bis 100

```
int i;
for(i=1;i<=10;i=i+1)
    System.out.println(i*i);
```

1. Durchlauf

Setze i=10

Ausgabe von 10\*10

Da i=11, ist der Endwert erreicht! Mache also hinter der Wiederholungsanweisung weiter!

Ausgabe insgesamt:

1	36
4	49
9	64
16	81
25	100



# Beispiel Id

Das Programm wird mit F7(schrittweise) gestartet. Dabei bekommt man die Werte der Variablen unten angezeigt:

```
int i;  
  
for (i=1;i<=10;i=i+1)  
    System.out.println(i*i);
```

## 1. Schleifendurchlauf:

Parameter und lokale Variablen

```
+ args = instance of java.lang.String[]  
  i = 1
```

```
int i;  
  
for (i=1;i<=10;i=i+1)  
    System.out.println(i*i);
```

## 2. Schleifendurchlauf:

Parameter und lokale Variablen

```
+ args = instance of java.lang.String[]  
  i = 2
```



# Beispiel II

**Ausgabe der Quadratzahlen 100 bis 1**

```
for (int i=10; i>=1; i--)  
    System.out.println(i*i);
```

**Ausgabe  
insgesamt:**

<b>100</b>	<b>25</b>
<b>81</b>	<b>16</b>
<b>64</b>	<b>9</b>
<b>49</b>	<b>4</b>
<b>36</b>	<b>1</b>

# Vergleich der drei Wdhlg.s.anweisungen

Gleichwertig sind folgende Alternativen:

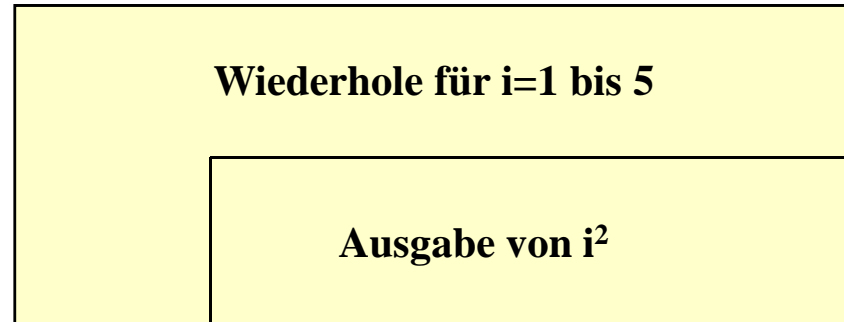
```
for(int i=1;i<=10;i++)  
System.out.println(i*i);
```

```
int i=1;  
while(i<=10)  
System.out.println(i*i++);
```

```
int i=1;  
do {  
System.out.println(i*i++);  
} while(i<=10);
```



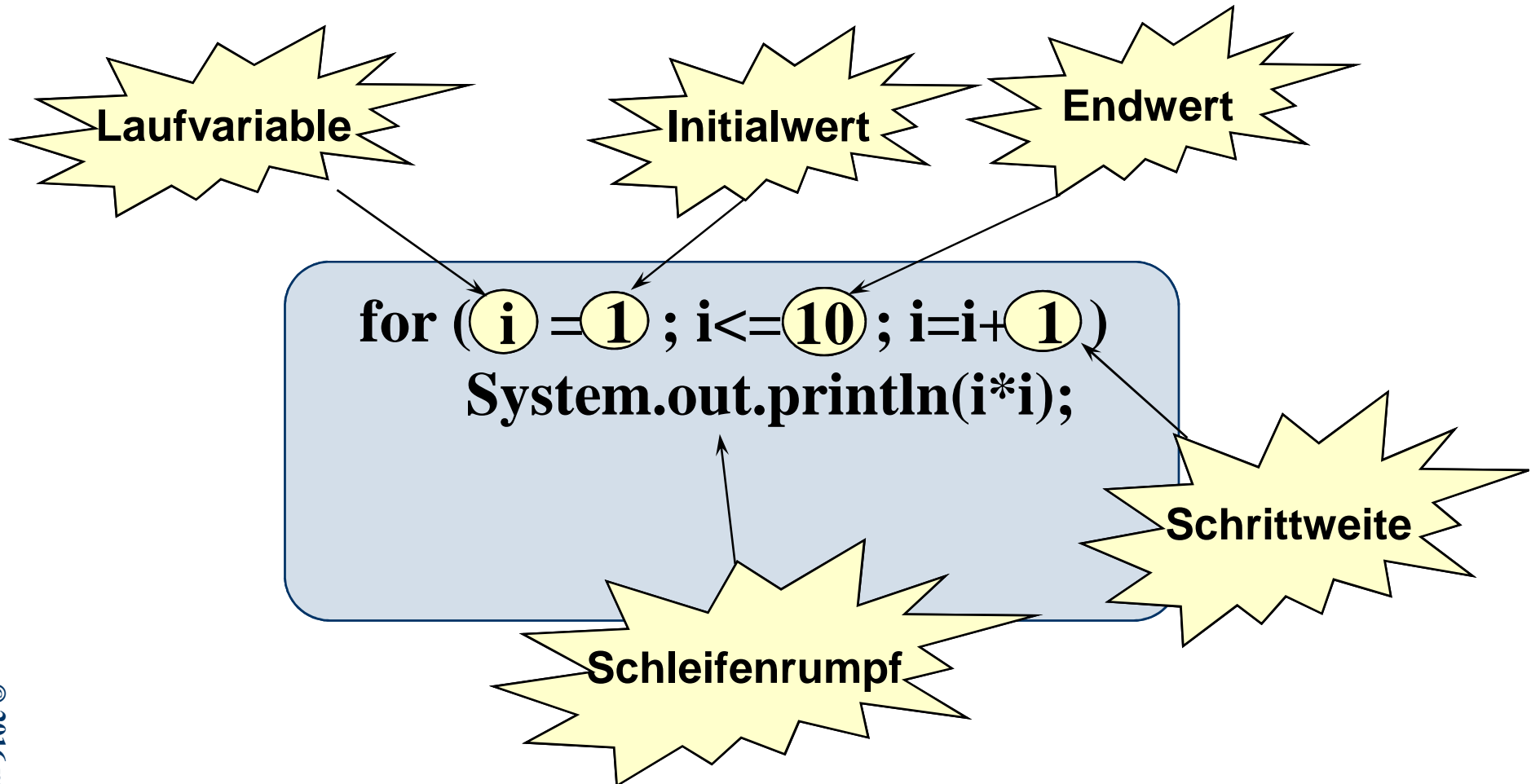
# For-Wdhgs.anw.: Struktogramm







# for-Wdhlg.s.anw.: Begriffe





# Umrechnung Celsius-Fahrenheit

```
double startWert;  
startWert=Keyboard.  
double celsius, fahrenheit;  
for (celsius=startWert;celsius<=40;celsius=celsius+5) {  
    fahrenheit = (9/5)*celsius+32;  
    System.out.println(celsius+"°C = "+fahrenheit+"°F");  
}
```



# Multiplikationstabelle

Multiplikationstabelle für das kleine 1x1:

```
int i,j;  
for(i=1;i<=3;i++){  
    for(j=1;j<=3;j++){  
        System.out.print(i+"*"+j+"="+i*j+" ");  
        System.out.println();  
    }  
}
```

C:\Windows\system32\cmd.exe

```
1*1=1 1*2=2 1*3=3  
2*1=2 2*2=4 2*3=6  
3*1=3 3*2=6 3*3=9
```

->Einmaleins für 14,15,16,17



# Multiplikationstabelle II

Zeile ausgeben

```
int i,j;  
for(i=1;i<=3;i++){  
    for(j=1;j<=3;j++)  
        System.out.print(i+"*"+j+"="+i*j+" ");  
    System.out.println();  
}
```

Zeilenumbbruch  
ausgeben(Schreibposition  
wandert in die nächste Zeile)



# Matrizen(Matrix) I

$$\mathbf{A}_{(m,n)} \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{pmatrix}$$

$\xrightarrow{\quad\quad\quad}$  n Spalten

$\downarrow$  m Zeilen

**n=m: quadratische Matrix**

**m: Zeilenindex**

**n: Spaltenindex**

**Der Index in „x“-Richtung kommt an zweiter Position!!!**



# Matrizen(Matrix) II

Korrelationsmatrix VB VarioZins Garant Express 2.0

	BASF	Coca Cola	Daimler	Dt. Börse	Dt. Post	Dt. Telekom	Microsoft	Nokia	Roche	RWE
BASF	1,00	0,90	0,80	0,49	0,68	0,41	0,81	0,34	0,36	0,42
Coca Cola	0,90	1,00	0,88	0,69	0,82	0,58	0,86	0,54	0,51	0,59
Daimler	0,80	0,88	1,00	0,83	0,89	0,78	0,79	0,75	0,61	0,75
Dt. Börse	0,49	0,69	0,83	1,00	0,93	0,84	0,70	0,95	0,77	0,91
Dt. Post	0,68	0,82	0,89	0,93	1,00	0,83	0,78	0,87	0,76	0,87
Dt. Telekom	0,41	0,58	0,78	0,84	0,83	1,00	0,61	0,87	0,75	0,84
Microsoft	0,81	0,86	0,79	0,70	0,78	0,61	1,00	0,60	0,69	0,71
Nokia	0,34	0,54	0,75	0,95	0,87	0,87	0,60	1,00	0,82	0,94
Roche	0,36	0,51	0,61	0,77	0,76	0,75	0,69	0,82	1,00	0,90
RWE	0,42	0,59	0,75	0,91	0,87	0,84	0,71	0,94	0,90	1,00



# Rechteck aus Sternen

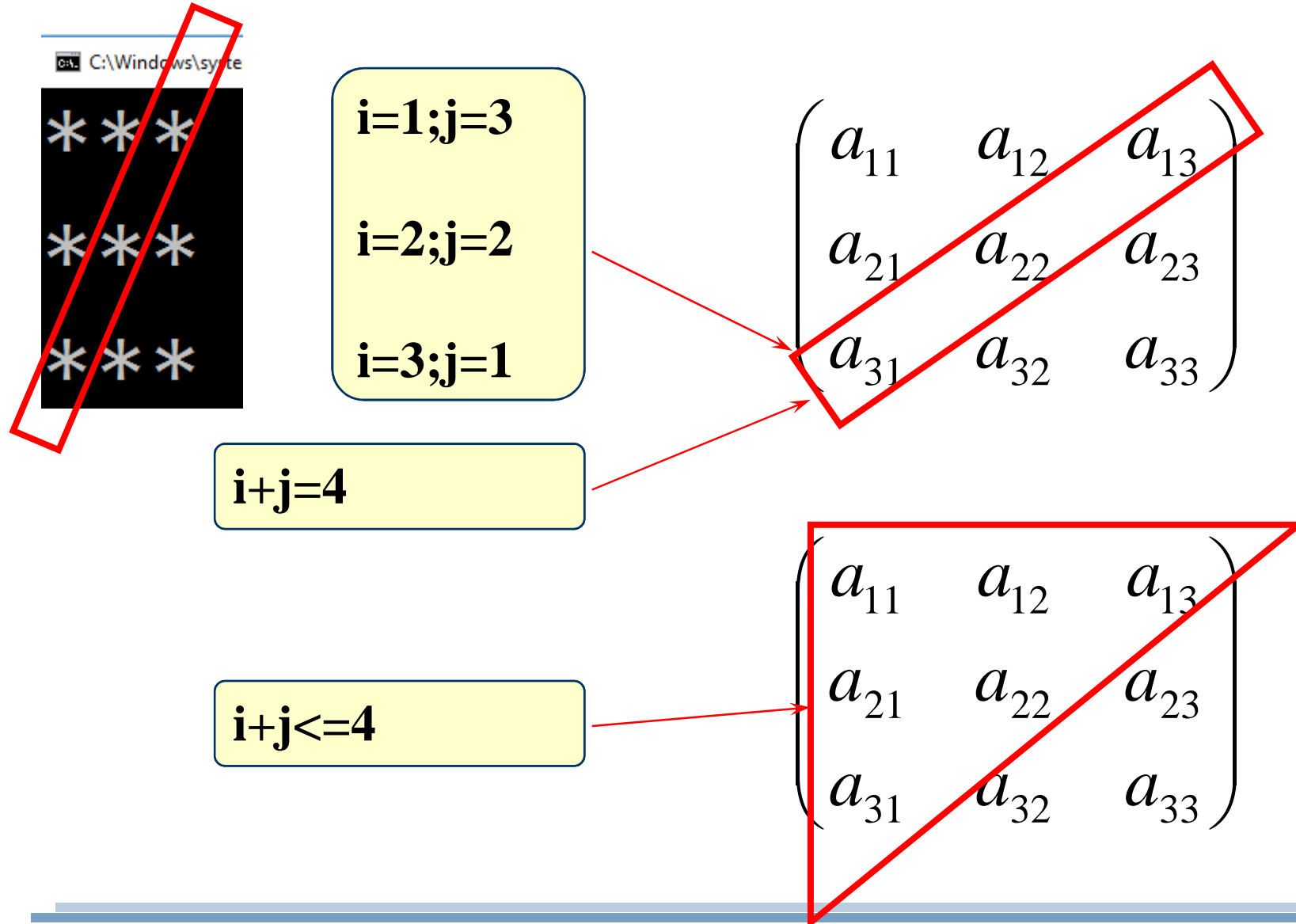
Legt die Anzahl an Zeilen  
und Spalten fest

```
int i,j;  
int anzZeilen=3;  
int anzSpalten=3;  
for(i=1;i<=anzZeilen;i++){  
    for(j=1;j<=anzSpalten;j++){  
        System.out.print("*");  
        System.out.println();  
    }  
}
```

Zeichnet eine Zeile an  
Sternen mit Zeilenumbruch



# Dreieck aus Sternen links oben I







# Rechteck aus Sternen links oben II

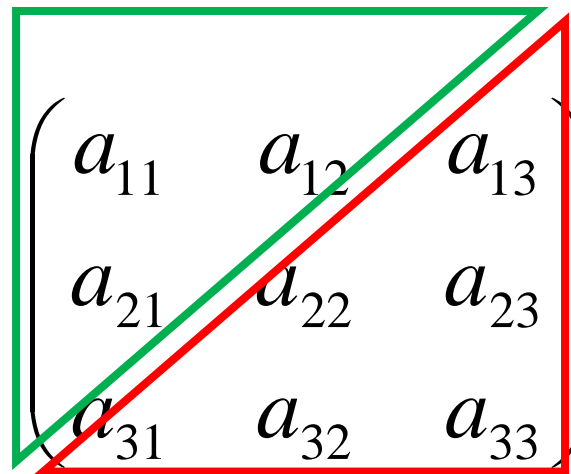
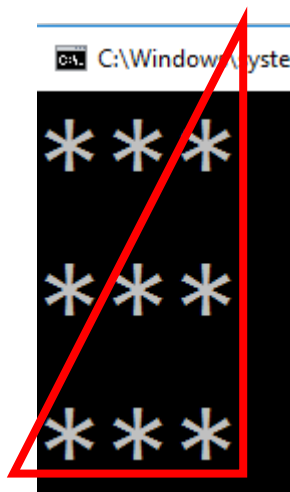
```
int i,j;  
int anzZeilen=3;  
int anzSpalten=3;  
for(i=1;i<=anzZeilen;i++){  
  for(j=1;j<=anzSpalten;j++){  
    if (i+j<=4)  
      System.out.print("*");  
    System.out.println();  
  }  
}
```

Gibt nur die Sterne mit  
Summe der Indizes kleiner  
als 4 an



# Dreieck aus Sternen rechts unten

sonst:->  
Leerzeichen



$i+j \geq 4$   
-> Sterne



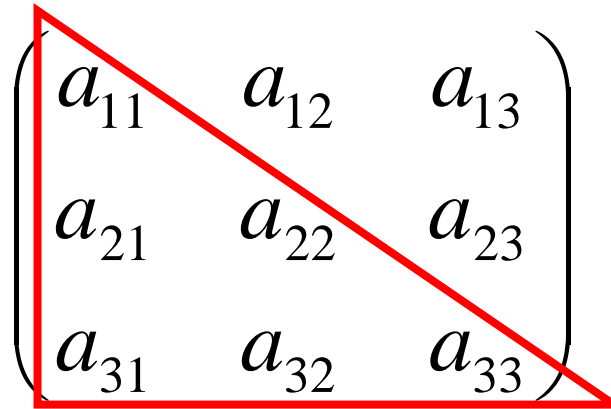
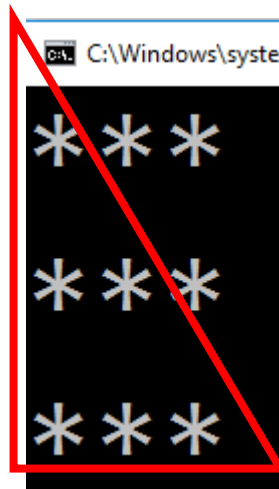
# Rechteck aus Sternen rechts unten II

```
int i,j;
int anzZeilen=3;
int anzSpalten=3;
for(i=1;i<=anzZeilen;i++){
  for(j=1;j<=anzSpalten;j++){
    if (i+j>=4)
      System.out.print("*");
    else
      System.out.print(" ");
    System.out.println();
  }
}
```

Wenn Summe der Indizes  
größer als 4,  
dann Sterne ausgeben,  
sonst Leerzeichen ausgeben!



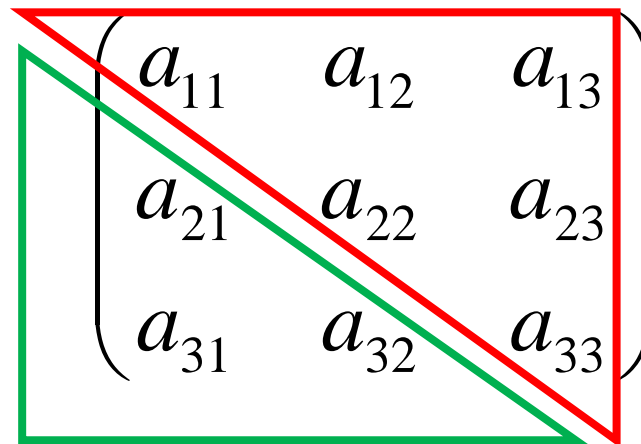
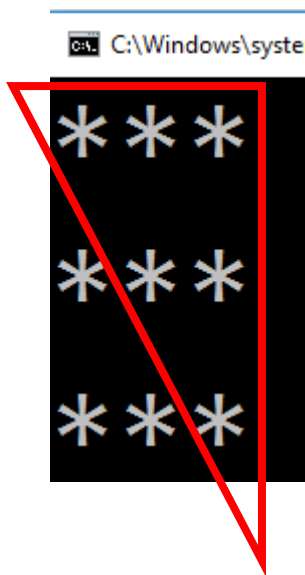
# Dreieck aus Sternen links unten



**$i \geq j$**   
**-> Sterne**



# Dreieck aus Sternen rechts oben



$i \leq j$   
 -> Sterne  
 sonst:  
 Leerzeichen



# „Baum“ aus Sternen

```
int i,j;  
int AnzSterne=11;  
for ( i=1; i<=AnzSterne;i=i+2){  
    for (j=(AnzSterne-i)/2; j>=0;j=j-1)  
        System.out.print(" ");  
    for (j=1; j<=i;j++)  
        System.out.print("*");  
    for (j=(AnzSterne-i)/2; j>=0;j=j-1)  
        System.out.print(" ");  
    System.out.println("");  
}
```



# Vergleich der drei Schleifen

```
while <Bedingung>  
.....
```

```
for  
.....
```

```
.....  
while <Bedingung>
```

Die Anzahl der Durchläufe ist **nicht** vorher bekannt

Die Anzahl der Durchläufe ist schon vorher bekannt; dieser Wert kann in einer Variablen stehen!

Die Anzahl der Durchläufe ist **nicht** vorher bekannt

Es ist möglich, dass die Schleife **kein einziges Mal** durchlaufen wird

Es ist möglich, dass die Schleife **kein einziges Mal** durchlaufen wird

Die Schleife wird **mindestens einmal** durchlaufen