

Kontrollstrukturen

Dr. Philipp Wendler

Zentralübung zur Vorlesung

„Einführung in die Informatik: Programmierung und Softwareentwicklung“

<https://www.sosy-lab.org/Teaching/2018-WS-InfoEinf/>

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
  
x = y;  
y = x;
```

Welchen Wert haben die Variablen x und y am Ende des Codefragments?

- $x = 2, y = 2$
- $x = 2, y = 4$
- $x = 4, y = 2$
- $x = 4, y = 4$

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:

x 2

Codezeile: `int x = 2;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:

x

2

y

4

x

2

Codezeile: `int x = 2;` `int y = 4;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:

x

2

y

4

x

2

y

4

x

--

Codezeile: `int x = 2;` `int y = 4;` `x = y;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

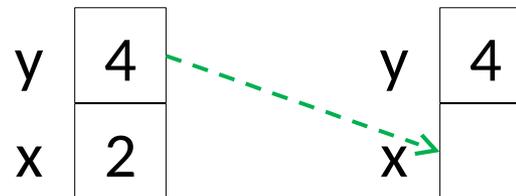
```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:



Codezeile: `int x = 2;` `int y = 4;` `x = y;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

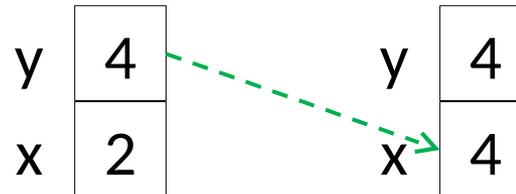
```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:



Codezeile: `int x = 2;` `int y = 4;` `x = y;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

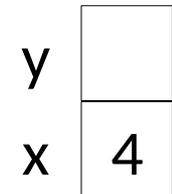
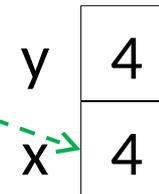
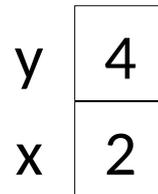
```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:



Codezeile: `int x = 2;` `int y = 4;` `x = y;` `y = x;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

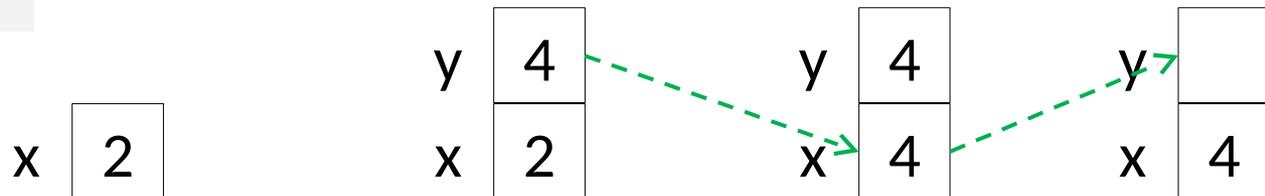
```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:



Codezeile: `int x = 2;` `int y = 4;` `x = y;` `y = x;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

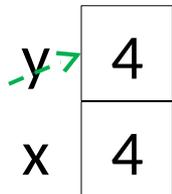
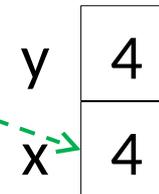
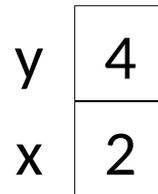
```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:



Codezeile: `int x = 2;` `int y = 4;` `x = y;` `y = x;`

Aufgabe 1a: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

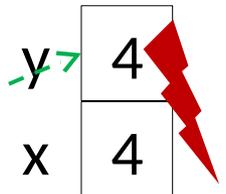
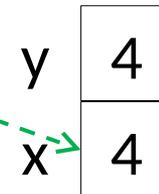
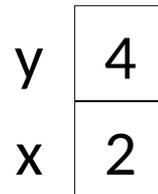
```
int x = 2;
```

```
int y = 4;
```

```
x = y;
```

```
y = x;
```

Veränderung
des Speichers:



Codezeile: `int x = 2;` `int y = 4;` `x = y;` `y = x;`

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```

Veränderung
des Speichers:

x 2

CZ: `int x = 2;`

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```

Veränderung
des Speichers:

x

2

y

4

x

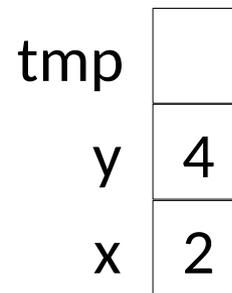
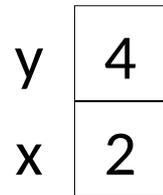
2

CZ: `int x = 2; int y = 4;`

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```



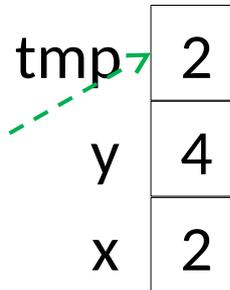
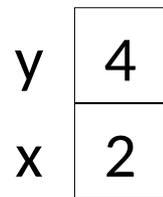
Veränderung
des Speichers:

CZ: `int x = 2; int y = 4; int tmp = x;`

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```



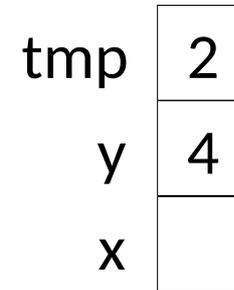
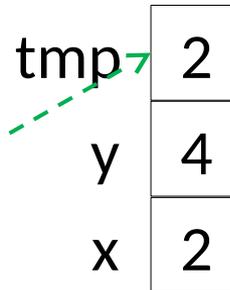
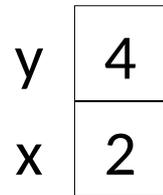
Veränderung
des Speichers:

CZ: `int x = 2; int y = 4; int tmp = x;`

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```



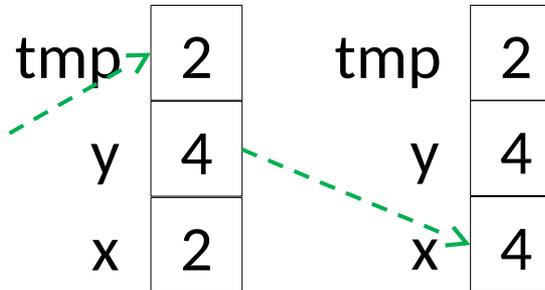
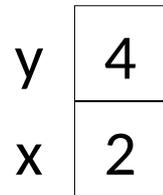
Veränderung
des Speichers:

CZ: `int x = 2; int y = 4; int tmp = x; x = y;`

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```



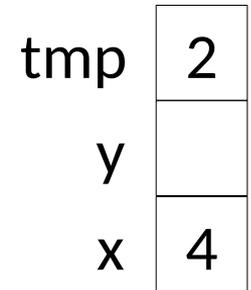
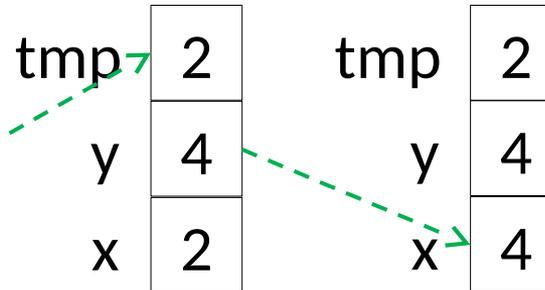
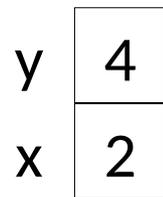
Veränderung
des Speichers:

CZ: `int x = 2; int y = 4; int tmp = x; x = y;`

Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```



Veränderung
des Speichers:

CZ: `int x = 2; int y = 4; int tmp = x; x = y; y = tmp;`

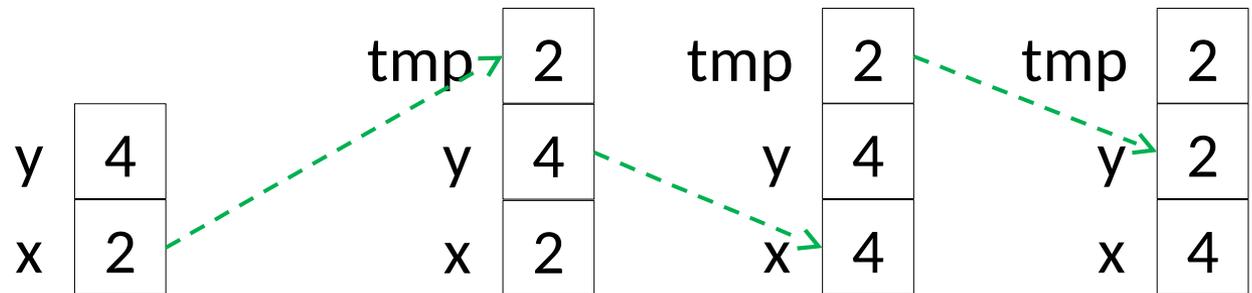
Aufgabe 1b: Zahlen vertauschen

Gegeben seien zwei Variablen x und y vom Typ `int`. Vertausche die Werte dieser beiden Variablen.

```
int x = 2;  
int y = 4;  
int tmp = x;  
x = y;  
y = tmp;
```



Veränderung
des Speichers:



CZ: `int x = 2; int y = 4; int tmp = x; x = y; y = tmp;`

Fallunterscheidung in Java

IfStatement =

vom Typ **boolean**



```
"if" "(" Expression ")" Statement  
["else" Statement]
```

```
Expression = Variable | Value |  
            | Expression BinOp Expression  
            | UnOp Expression  
            | "(" Expression ")"
```

```
Statement = VariableDeclaration  
           | Assignment  
           | Block  
           | Conditional  
           | Iteration
```

Fallunterscheidung in Java

IfStatement =

vom Typ **boolean**



```
"if" "(" Expression ")" Statement  
["else" Statement]
```

```
Expression = Variable | Value |  
            | Expression BinOp Expression  
            | UnOp Expression  
            | "(" Expression ")"
```

```
Statement = VariableDeclaration  
           | Assignment  
           | Block  
           | Conditional  
           | Iteration
```

int x = 2;

Fallunterscheidung in Java

IfStatement =

vom Typ **boolean**



```
"if" "(" Expression ")" Statement  
["else" Statement]
```

```
Expression = Variable | Value |  
            | Expression BinOp Expression  
            | UnOp Expression  
            | "(" Expression ")"
```

```
Statement = VariableDeclaration  
           | Assignment  
           | Block  
           | Conditional  
           | Iteration
```

int x = 2;

x = 4;

Fallunterscheidung in Java

IfStatement =

vom Typ **boolean**



```
"if" "(" Expression ")" Statement  
["else" Statement]
```

```
Expression = Variable | Value |  
            | Expression BinOp Expression  
            | UnOp Expression  
            | "(" Expression ")"
```

```
Statement = VariableDeclaration  
           | Assignment  
           | Block  
           | Conditional  
           | Iteration
```

int x = 2;

x = 4;

{ ... }

Einfache Fallunterscheidungen

- `int` alter = ?; // ? geeignet ersetzen
`boolean` volljaehrig = **false**;

```
if (alter >= 18) volljaehrig = true;
```

- `int` alter = ?; // ? geeignet ersetzen
`boolean` volljaehrig;

```
if (alter >= 18) volljaehrig = true;  
else volljaehrig = false;
```

Aufgabe 2a: Mehrere Statements

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18)
    volljaehrig = true;
    anrede = "Sie";
```

Welchen Wert haben die Variablen `volljaehrig` und `anrede` am Ende dieses Codefragments?

- `volljaehrig = false,`
`anrede = "du"`
- `volljaehrig = false,`
`anrede = "Sie"`
- `volljaehrig = true,`
`anrede = "du"`
- `volljaehrig = true,`
`anrede = "Sie"`
- Syntaktischer Fehler

Aufgabe 2a: Mehrere Statements

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18)
    volljaehrig = true;
    anrede = "Sie";
```

Welchen Wert haben die Variablen `volljaehrig` und `anrede` am Ende dieses Codefragments?

volljaehrig: false
anrede: "Sie"

Wegen:

```
IfStatement =
    "if" "(" Expression ")"
        Statement
    ["else" Statement]
```

Aufgabe 2b: Mehrere Statements

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18) {
    volljaehrig = true;
    anrede = "Sie";
}
```

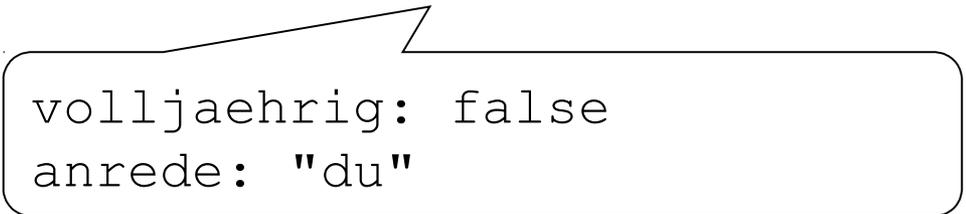
Welchen Wert haben die Variablen `volljaehrig` und `anrede` am Ende dieses Codefragments?

Aufgabe 2b: Mehrere Statements

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18) {
    volljaehrig = true;
    anrede = "Sie";
}
```

Welchen Wert haben die Variablen volljaehrig und anrede am Ende dieses Codefragments?



```
volljaehrig: false
anrede: "du"
```

Aufgabe 3a: Mehrere Statements pro Fall

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18)
    volljaehrig = true;
    anrede = "Sie";

else
    volljaehrig = false;
    anrede = "du";
```

Welchen Wert haben die Variablen `volljaehrig` und `anrede` am Ende dieses Codefragments?

- `volljaehrig = false,`
`anrede = "du"`
- `volljaehrig = false,`
`anrede = "Sie"`
- `volljaehrig = true,`
`anrede = "du"`
- `volljaehrig = true,`
`anrede = "Sie"`
- Syntaktischer Fehler

Aufgabe 3a: Mehrere Statements pro Fall

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18)
    volljaehrig = true;
    anrede = "Sie";

else
    volljaehrig = false;
    anrede = "du";
```

Welchen Wert haben die Variablen `volljaehrig` und `anrede` am Ende dieses Codefragments?

Achtung syntaktischer Fehler:

- Nach **if** (Expression) muss **ein** Statement folgen.
- Mehrere Assignments sind **kein** Statement!

Wegen: IfStatement =
 "**if**" "(" Expression ")"
 Statement
 ["**else**" Statement]

Aufgabe 3b: Mehrere Statements pro Fall

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18) {
    volljaehrig = true;
    anrede = "Sie";
}
else {
    volljaehrig = false;
    anrede = "du";
}
```

Welchen Wert haben die Variablen `volljaehrig` und `anrede` am Ende dieses Codefragments?

Aufgabe 3b: Mehrere Statements pro Fall

```
int alter = 4;
boolean volljaehrig
    = false;
String anrede = "du";

if (alter >= 18) {
    volljaehrig = true;
    anrede = "Sie";
}
else {
    volljaehrig = false;
    anrede = "du";
}
```

Welchen Wert haben die Variablen volljaehrig und anrede am Ende dieses Codefragments?



volljaehrig: false
anrede: "du"

Aufg 4a: Geschachtelte Fallunterscheidungen

Wir wollen überdurchschnittliche Studenten loben und “Durchfaller” zum Lernen anregen.

```
boolean durchgefallen = ?
    // ? geeignet ersetzen
double note = ?
    // ? geeignet ersetzen
String nachricht;
if (!durchgefallen)
    if (note <= 1.3)
        nachricht = "Sehr gut!";

else
    nachricht = "Lern mehr!";
```

Welche Studenten erhalten die Nachricht "Lern mehr"?

- Alle Studenten
- Alle Studenten mit Note 1.3 oder besser
- Alle Studenten, die mit Note 1.7 oder schlechter bestanden haben
- Alle durchgefallenen Studenten

Aufg 4a: Geschachtelte Fallunterscheidungen

Wir wollen überdurchschnittliche Studenten loben und “Durchfaller” zum Lernen anregen.

```
boolean durchgefallen = ?
    // ? geeignet ersetzen
double note = ?
    // ? geeignet ersetzen
String nachricht;
if (!durchgefallen)
    if (note <= 1.3)
        nachricht = "Sehr gut!";

else
    nachricht = "Lern mehr!";
```

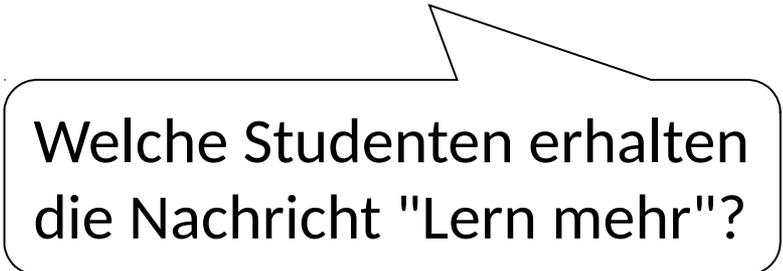
Welche Studenten erhalten die Nachricht "Lern mehr"?

Alle Studenten, die mit einer Note von 1.7 oder schlechter bestanden haben. Der `else`-Zweig gehört immer zum nächsten `if` im selben Block.

Aufg 4b: Geschachtelte Fallunterscheidungen

Wir wollen überdurchschnittliche Studenten loben und “Durchfaller” zum Lernen anregen.

```
boolean durchgefallen = ?
    // ? geeignet ersetzen
double note = ?
    // ? geeignet ersetzen
String nachricht;
if (!durchgefallen) {
    if (note <= 1.3)
        nachricht = "Sehr gut!";
}
else
    nachricht = "Lern mehr!";
```



Welche Studenten erhalten die Nachricht "Lern mehr"?

Aufg 4b: Geschachtelte Fallunterscheidungen

Wir wollen überdurchschnittliche Studenten loben und “Durchfaller” zum Lernen anregen.

```
boolean durchgefallen = ?  
    // ? geeignet ersetzen  
double note = ?  
    // ? geeignet ersetzen  
String nachricht;  
if (!durchgefallen) {  
    if (note <= 1.3)  
        nachricht = "Sehr gut!";  
}  
else  
    nachricht = "Lern mehr!";
```

Welche Studenten erhalten die Nachricht "Lern mehr"?

Alle durchgefallenen Studenten

Aufg 4c: Geschachtelte Fallunterscheidungen

Wir wollen überdurchschnittliche Studenten loben und “Durchfaller” zum Lernen anregen.

```
boolean durchgefallen = ?  
    // ? geeignet ersetzen  
double note = ?  
    // ? geeignet ersetzen  
String nachricht;  
  
if (durchgefallen)  
    nachricht = "Lern mehr!";  
else  
    if (note <= 1.3)  
        nachricht = "Sehr gut!";
```

Welche Studenten erhalten die Nachricht "Lern mehr"?

- Alle Studenten
- Alle Studenten mit Note 1.3 oder besser
- Alle Studenten, die mit Note 1.7 oder schlechter bestanden haben
- Alle durchgefallenen Studenten

Aufg 4c: Geschachtelte Fallunterscheidungen

Wir wollen überdurchschnittliche Studenten loben und “Durchfaller” zum Lernen anregen.

```
boolean durchgefallen = ?
    // ? geeignet ersetzen
double note = ?
    // ? geeignet ersetzen
String nachricht;

if (durchgefallen)
    nachricht = "Lern mehr!";
else
    if (note <= 1.3)
        nachricht = "Sehr gut!";
```

Welche Studenten erhalten die Nachricht "Lern mehr"?

Alle durchgefallenen Studenten

Aufg 4d: Geschachtelte Fallunterscheidungen

Wir wollen allen Studenten eine entsprechende Nachricht schicken.

```
... // analog zu vorher
if (durchgefallen)
    nachricht = "Lern mehr!";
else if (note <= 1.3)
    nachricht = "Sehr gut!";
else if (note <= 2.3)
    nachricht = "Gut!";
else if (note <= 3.3)
    nachricht = "Befriedigend!";
else nachricht = "Ausreichend!";
```

Praktischer Tipp für Fallunterscheidungen

Immer einen Block nach `if` und `else` verwenden, um Missinterpretation zu vermeiden!

```
if (durchgefallen) {  
    nachricht = "Lern mehr!";  
}  
else if (note <= 1.3) {  
    nachricht = "Sehr gut!";  
}  
...  
else {  
    nachricht = "Ausreichend!";  
}
```