

# Üben 3: Debugger, Schleifen

Einführung in die Programmierung 1  
Wintersemester 22

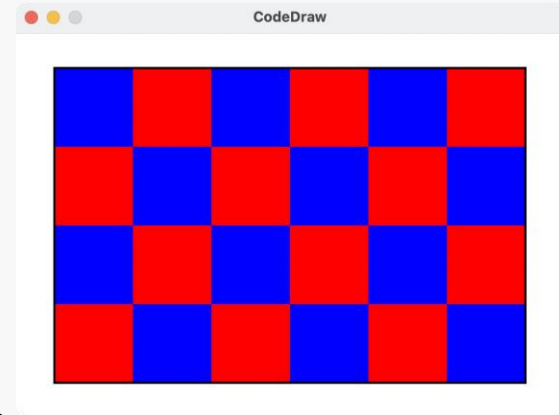


# Vorbereitung

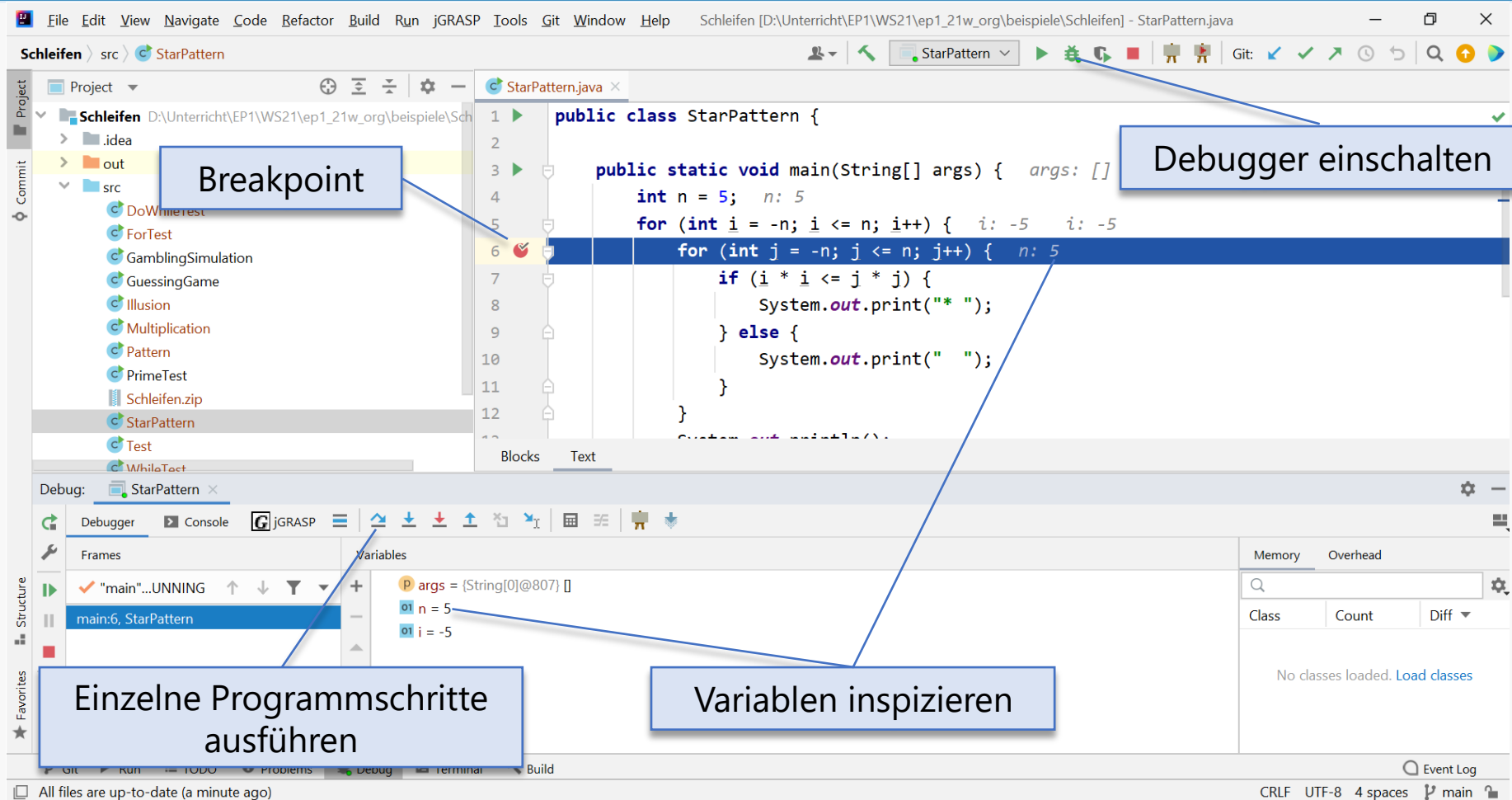
- TUWEL öffnen, IntelliJ-Projekt „Ueben3.zip“ herunterladen
- Projekt in IntelliJ öffnen
- Laden Sie das bearbeitete Projekt am Ende wieder in TUWEL hoch

# CodeDrawDebug (Aufgabe 1)

- Der gegebene Code soll die dargestellte Grafik erzeugen, enthält jedoch Fehler
- **Benutzen Sie den Debugger**, um die Fehler aufzuspüren und auszubessern:
  - Setzen Sie einen Breakpoint in Zeile 18
  - Gehen Sie Zeile für Zeile durch das weitere Programm
  - Beobachten Sie die Werte der Variablen
  - *Als Hilfestellung:*
    - Die korrekten x-Koordinaten der oberen linken Ecken der Quadrate sind: 35.5, 107, 178.5, 250, 321.5, 393
- Es dürfen keine Ausgaben mittels System.out erzeugt werden!



# Debugger



# Werte von Ausdrücken inspizieren

- Mauszeiger auf gewünschten Ausdruck (Operator) bewegen

The image displays two screenshots of a code editor interface, illustrating how to inspect the value of an expression by hovering over different operators.

**Top Screenshot:** The code is `int result = a + b * c; a: 2 b: 3 c: 5`. The mouse cursor is hovering over the multiplication operator `*`. A tooltip shows the value `15`, which is the result of `b * c` (3 \* 5). A callout box points to the tooltip with the text "Maus über \*".

**Bottom Screenshot:** The code is the same: `int result = a + b * c; a: 2 b: 3 c: 5`. The mouse cursor is now hovering over the addition operator `+`. A tooltip shows the value `17`, which is the result of `a + b * c` (2 + 15). A callout box points to the tooltip with the text "Maus über +".

# Passwords (Aufgabe 2)

- Schreiben Sie ein Programm, das überprüft, ob das eingegebene Passwort sicher ist
  - Lesen Sie das Passwort ein (mit einem Scanner)
  - Ein Passwort wird als sicher eingestuft, wenn es mindestens aus 10 Zeichen besteht und einen Großbuchstaben enthält

*Hinweis:* Verwenden Sie die String-Methoden `length()` und `charAt(int index)`

Geben Sie ein Passwort ein:

*afDpMdMxGp*

Das Passwort ist sicher

Geben Sie ein Passwort ein:

*hallowelt*

Das Passwort ist nicht sicher

# Passwords (Aufgabe 2) – 1. Erweiterung

- Erweitern Sie das Programm so, dass ein Passwort als sicher eingestuft wird, wenn folgende Bedingungen erfüllt sind
  - Das Passwort besteht aus mindestens 10 Zeichen
  - Das Passwort enthält mindestens einen Großbuchstaben, einen Kleinbuchstaben und eine Ziffer

Geben Sie ein Passwort ein:

*kxNpD4MaB2fG*

Das Passwort ist sicher

Geben Sie ein Passwort ein:

*afDpMdMxGp*

Das Passwort ist nicht sicher

# Passwords (Aufgabe 2) – 2. Erweiterung

- Erweitern Sie das Programm so, dass es ausgibt, warum ein Passwort als unsicher eingestuft wird
  - Wenn das Passwort aus mehreren Gründen als unsicher betrachtet wird, müssen **alle** nicht geltenden Bedingungen mitgeteilt werden

Geben Sie ein Passwort ein:

*halLo*

Das Passwort ist nicht sicher

- \* Das Passwort ist zu kurz
- \* Das Passwort enthält keinen Großbuchstaben
- \* Das Passwort enthält keine Ziffer



# Passwords (Aufgabe 2) – 3. Erweiterung (optional)

- Erweitern Sie das Programm so, dass ein Passwort als unsicher eingestuft wird, wenn es eine Folge desselben Zeichens enthält, deren Größe 4 oder mehr beträgt
  - Die gefundene, ungültige Zeichenfolge muss in der ausgegebenen Fehlermeldung enthalten sein
  - Bei mehreren solcher Zeichenfolgen kann nur die erste ausgegeben werden

Geben Sie ein Passwort ein:

*af5Gl1111p*

Das Passwort ist nicht sicher

\* Die Sequenz 1111 darf nicht im Passwort vorkommen

Geben Sie ein Passwort ein:

*AAAAA6pm5P*

Das Passwort ist nicht sicher

\* Die Sequenz AAAAA darf nicht im Passwort vorkommen

# Triangles (Aufgabe 3)

- Implementieren Sie das folgende Programm:
  - Lesen Sie eine Zahl ein
  - Geben Sie die eingelesene Anzahl an Zeilen aus
    - Die erste Zeile enthält eine 1, die zweite Zeile zwei mal die 2, die dritte drei mal die 3, usw.

Eingabe:

0

Eingabe:

1

1

Eingabe:

2

1

22

Eingabe:

4

1

22

333

4444

# Triangles (Aufgabe 3) – 1. Erweiterung

- Erweitern Sie Ihr Programm so, dass zwischen zwei Zahlen jeweils das Zeichen '|' ausgegeben wird
  - Die Anzahl an Zahlen pro Zeile bleibt unverändert

Eingabe:

0

Eingabe:

1

1

Eingabe:

2

1

2|2

Eingabe:

4

1

2|2

3|3|3

4|4|4|4

# Triangles (Aufgabe 3) – 2. Erweiterung (optional)

- Erweitern Sie Ihr Programm so, dass zwischen zwei Zahlen jeweils eine Zeile bestehend aus so vielen '-', wie die Zeile darüber Zeichen enthält, ausgegeben wird
  - Die Anzahl der Zeilen mit Zahlen bleibt unverändert
  - Das Programm soll auch mit mehrstelligen Zahlen richtig funktionieren

Eingabe:

4

1

-

2|2

---

3|3|3

-----

4|4|4|4

Eingabe:

11

...

9|9|9|9|9|9|9|9|9

-----

10|10|10|10|10|10|10|10|10|10

-----

11|11|11|11|11|11|11|11|11|11|11

# ConsolePattern (Aufgabe 4)

- Implementieren Sie das folgende Programm:
  - Lesen Sie einen Integer n ein (mittels Scanner)
  - Geben Sie n Zeilen aus, wobei jede Zeile genau n Punkte '.' enthält
  - Der Lerneffekt ist größer, wenn Sie das Programm selber schreiben und nicht aus den Vorlesungsfolien kopieren

Größe des Quadrats:

1

.

Größe des Quadrats:

2

..

..

Größe des Quadrats:

5

.....

.....

.....

.....

.....

# ConsolePattern (Aufgabe 4) – 1. Erweiterung

- Erweitern Sie das Programm wie folgt:
  - Geben Sie in den Ecken das Zeichen '+' aus
  - Geben Sie in der ersten und letzten Zeile das Zeichen '-' aus
  - Geben Sie in der ersten und letzten Spalte das Zeichen '|' aus
  - Geben Sie an den anderen Stellen des Quadrats das Zeichen '.' aus

Größe des Quadrats:

1

+

Größe des Quadrats:

2

++

++

Größe des Quadrats:

5

+---+

|...|

|...|

|...|

+---+

# ConsolePattern (Aufgabe 4) – 2. Erweiterung (optional)

- Erweitern Sie das Programm wie folgt:
  - Beim Einlesen prüfen Sie, dass die eingegebene Größe eine ungerade, positive Zahl ist
    - Wenn nicht, Fehlermeldung anzeigen und das Einlesen wiederholen, bis die obere Bedingung gilt
  - Geben Sie in der Hauptdiagonale das Zeichen `'\'` aus
  - Geben Sie in der Gegendiagonale das Zeichen `'/'` aus
  - Geben Sie das Zeichen `'x'` dort aus, wo sich die Diagonalen treffen

```
Größe des Quadrats:  
3  
+--+  
|x|  
+--+
```

```
Größe des Quadrats:  
5  
+----+  
|\./|  
|.x.|  
|/.\|  
+----+
```

```
Größe des Quadrats:  
7  
+-----+  
|\...\|  
|.\/.\/|  
|.x.x.|  
|/.\.\|  
|/...\|  
+-----+
```

**Danke für's mitmachen**

Bitte Projekt zippen und in TUWEL hochladen

**Bis nächste Woche**