

PGdP Woche #2 - Kontrollstrukturen

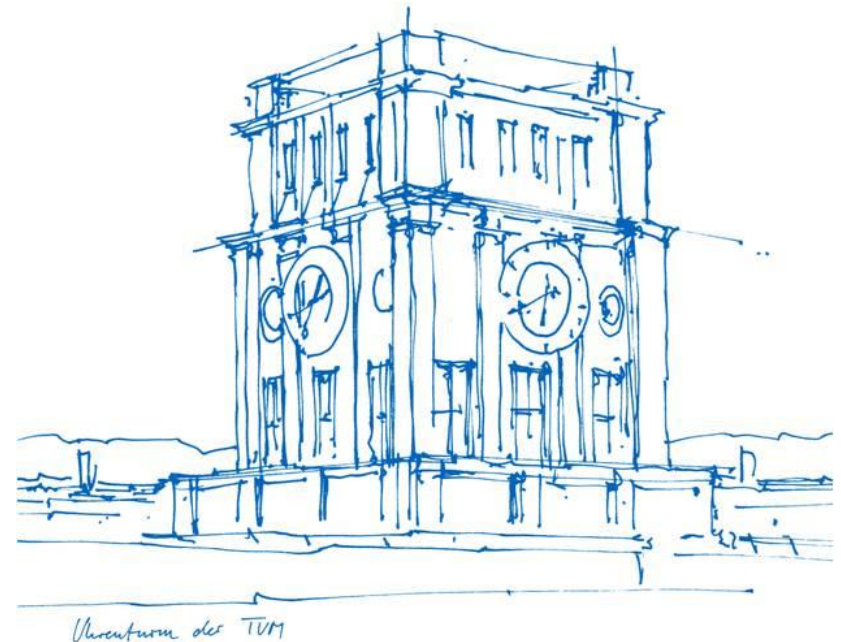
Sebastian Oßner – ossner@in.tum.de

Technische Universität München

Garching, 28. Oktober 2019

Ablauf:

1. Überblick Kontrollstrukturen
2. P01 – Summieren
3. P04 – 3 und 7
4. P02 – Primfaktorzerlegung
5. P03 – Karnickelpopulation



Kontrollstrukturen

Arten von Kontrollstrukturen

Loops:

```
// For-Loop
for (int i = 0; i < x; i++) {
    doStuff();
}

// While-Loop
while (x > 2) {
    doStuff();
}

// Do-While Loop
do {
    doStuff();
} while (x > 2);
```

Conditional Statements:

```
if (x > 2) {
    doStuff();
} else if (x < 2) {
    doStuff();
} else {
    doStuff();
}
```

P01 - Summieren

- Nutzer nach Input fragen BIS etwas passiert
- Zahlen nach Input jeweils aufsummieren
- Bis 0 eingegeben wird
- Ergebnis ausgeben

Methoden:

- Input bekommen: `readInt()`
- Auf `commandLine` ausgeben: `write()`

Beispiel read/write:

- `int x = readInt();`
- `write(x);`

Anzahl verwendeter Kontrollstrukturen: 1

P04 – 3 und 7

- Nutzer nach einer Zahl n fragen
- Summe aller Zahlen x \leq n, die durch 3 oder 7 teilbar sind berechnen
- Summe ausgeben
- Ungültiger Input für: n $<$ 0

Methoden:

- Input bekommen: `readInt()`
- Auf `commandLine` ausgeben: `write()`

Anzahl verwendeter Kontrollstrukturen: 3

P02 - Primfaktorzerlegung

- Nutzer nach einer Zahl fragen
- Ungültiger Input bei $n \leq 1$
- Zahl in Primfaktoren zerlegen und Primfaktoren mit „ “ getrennt ausgeben

Methoden:

- | | |
|--|---------------------------------|
| - Input bekommen: | <code>readInt()</code> |
| - Auf commandLine ausgeben (Ohne Zeilenumbruch): | <code>writeConsole()</code> |
| - Auf commandLine Zeilenumbruch ausgeben: | <code>writeLineConsole()</code> |

Anzahl verwendeter Kontrollstrukturen: 4

P03 - Karnickelpopulation

- Am Anfang 1 Paar Karnickel
- Karnickel sterben nach 3 Monaten
- Karnickel sind sofort geschlechtsreif
- Ein existierendes Paar bringt jeden Monat ein neues Paar auf die Welt
- Nutzer nach Anzahl der Monate fragen
- Anzahl Kaninchenpaare berechnen und ausgeben

Tipps:

- Nested loop (loop in einem anderen Loop)
- Array mit Population pro Generation

Anzahl verwendeter Kontrollstrukturen: 2