

5. Übungsblatt - Informatik 1 - Lösungsbeispiele

Aufgabe 1 (if, switch)

Schreiben Sie folgendes Programmfragment so um, dass if-else statt switch als Fallunterscheidung verwendet wird. a und b sind int-Variablen.

```
switch (a * 5 - b) {  
case -2:  a = 1;  
case 3:   b = b + a;  
         break;  
default:  b = a - 5;  
case 5:   a = b + 1;  
}
```

Lösungsvorschlag:

```
int c = a * 5 - b;  
if (c == -2) {  
    a = 1;  
} else if (c == 3 || c == -2) {  
    b = b + a;  
} else {  
    if (c != 5) {  
        b = a - 5;  
    }  
    a = b + 1;  
}
```

Aufgabe 2 (if)

Gegeben seien drei int-Variablen a, b und c. Schreiben Sie ein Programm, welches die Werte dieser Variablen so ändert, dass am Ende des Programms die Werte aufsteigend sortiert sind, also $a \leq b \leq c$ gilt. Wenn die Werten zu Beginn beispielsweise $a = 5$, $b = -3$, $c = 2$ sind, dann sollen die Variablen am Ende folgenden Zustand haben: $a = -3$, $b = 2$, $c = 5$.

Lösungsvorschlag:

```
int a = 5;  
int b = -2;  
int c = 3;  
  
int t;
```

```

if (a >= b) {
    t = a;
    a = b;
    b = t;
}
// ab hier gilt immer: a < b
if (b >= c) {
    t = b;
    b = c;
    c = t;
}
// Das Maximum ist auf jeden Fall in c gewandert
// ab hier gilt immer: a < b
if (a >= b) {
    t = a;
    a = b;
    a = t;
}
// a <= b <= c

```

Aufgabe 3 (if, Bitweise Operatoren)

Überlegen Sie sich, wie bei Addition zweier ganzzahliger Variablen in Java ein Überlauf oder Unterlauf entdeckt werden kann und einen entsprechenden Fehlertext ausgibt. Erklären Sie Ihre Lösung. Implementieren Sie Ihre Lösung als Programm. Das Programm soll für jeden ganzzahligen Datentyp funktionieren.

Hinweis: Ist das Vorzeichen der Variablen verschieden, dann kann kein Über- oder Unterlauf entstehen. Das Programm kann mit Hilfe der Bitweisen Operatoren kurz gehalten werden.

Lösungsvorschlag:

Sind die Vorzeichen von a und b gleich, dann ändert sich bei einem Über- oder Unterlauf das Vorzeichen der Summe. Ob die Vorzeichenbits zweier Zahlen im 2er-Komplement gleich sind, kann mit einer XOR-Verknüpfung der Vorzeichenbits (oder aller Bits) überprüft werden: Ist das Ergebnis 0 (oder die Zahl positiv), dann waren die Bits gleich und deswegen auch die Vorzeichen.

```

long a = 1000L;
long b = 1111L;
long summe = a + b;
if ( ((a ^ b) > 0) // Vorzeichenbits sind gleich
    && ((a ^ summe) < 0) // Vorzeichenbits sind unterschiedlich
    ) {
    System.out.println("Ueberlauf_/_\Unterlauf");
}

```