
Programmierstarthilfe SS 2010
Fakultät für Ingenieurwissenschaften und Informatik

1. Blatt

Für die Woche vom 26.04. bis zum 30.04.2010 (KW 17)

Organisatorisches

Im Web unter <http://www.uni-ulm.de/in/programmierstarthilfe.html>

An folgenden Terminen findest Du uns im O28-Pool:

Montag 10 - 12	Montag 12 - 14	Montag 14 - 16	Mittwoch 16 - 18
-------------------	-------------------	-------------------	---------------------

Die Programmierstarthilfe beginnt bei Null, wir gehen von keinem Vorwissen aus.

Jede Woche gibt es ein neues Blatt, das wir an den Terminen austeilen und das auch auf unserer Webseite verfügbar ist.

Wir üben im PC-Pool im O28. Es gibt jede Woche mehrere Termine zu denen ihr kommen könnt und es wird immer am selben Blatt gearbeitet.

Zu jedem Termin sind Tutoren da, die dich bei der Lösung der Aufgaben beraten. Für einen optimalen Lernerfolg empfehlen wir dir zuerst zu überlegen wie du die Aufgabe lösen willst, bei längeren Lösungen eine kleine Skizze auf Papier anzufertigen und erst dann das eigentliche Programm zu schreiben.

Diese Woche beginnen wir mit einem kleinen ersten Programm und elementaren Datentypen. Dazu verwenden wir nur einen Texteditor (Notepad++) und die Kommandozeile. Wir rechnen mit Wahrheitswerten und Strings und machen Typecasts. Dieses Blatt bezieht sich auf Abschnitt 0.7 aus der Einleitung des Skripts und auf das Kapitel *Datentypen*.

0.1 Account im SGI-Pool

Beantrage einen Account im Pool, dazu logge dich an einem beliebigen Rechner als `antrag` ohne Passwort ein und gib deine Daten an. Unterschreibe den Antrag und bringe ihn in der Servicegruppe Informatik vorbei.

0.2 Notepad++

Java kann man mit jedem Texteditor programmieren, wir verwenden Notepad++. Bevor Du mit den Aufgaben startest solltest du die Zeilennummern aktivieren (unter Settings → Preferences → Editing → Display Line Numbers). Achte darauf, dass du deine Dateien immer auf deinem Homelaufwerk (H:) speicherst.

1 Aufgaben

In diesem Abschnitt finden sich immer die Aufgaben, die du bearbeiten sollst um die Konzepte dieses Blatts zu lernen.

1.1 Ein erstes Programm

Gib das folgende Programm ein, kompiliere es und führe es aus. Hierzu hilft dir Abschnitt 0.7 aus der Einleitung des Skripts.

```

1 public class Hello {
2     public static void main (String[] args) {
3         System.out.println("Hello World!");
4     }
5 }

```

Ändere das Programm ab, so dass dein eigener Name ausgegeben wird.

1.2 Variable 1

- a) Schreibe ein Programm, in dem eine Variable `a` vom Typ `int` deklariert wird, die dann den Wert 5 zugewiesen bekommt. Gib anschließend den Wert dieser Variable durch den Befehl `System.out.println(a);` aus.
- b) Deklariere eine zweite Variable `b` und weise ihr in der gleichen Zeile einen Wert zu. Gib mit `System.out.println(a+b);` die Summe der beiden Variablen aus.

1.3 Variable 2

Ändere das Programm von Aufgabe Variable 1 so ab, dass eine dritte Variable `c` vom Typ `int` deklariert und ihr der Wert der Summe `a + b` zugewiesen wird. Gib anschließend den Wert von `c` aus.

1.4 Variable 3

Ändere das Programm von Aufgabe Variable 2 so ab, dass es statt `int` den Datentyp `double` verwendet.

1.5 Boolean 1

Es gibt in Java einen besonderen Datentyp, der nur zwei Werte annehmen kann: `boolean`. Dieser `boolean` kann die Werte `true` oder `false` annehmen.

Schreibe ein Programm, das eine Variable vom Typ `boolean` deklariert, ihr den Wert `true` zuweist und danach den Wert der Variablen ausgibt.

1.6 Boolean 2

In Java gibt es die Möglichkeit Vergleiche zu machen. Das Ergebnis eines Vergleiches ist vom Typ `boolean`.

- a) Ändere das Programm der Aufgabe Boolean 1 so ab, dass der Variablen das Ergebnis des Vergleichs `5<=10` zugewiesen wird.
- b) Deklariere weitere Variablen, denen du jeweils den Wert eines beliebigen weiteren Vergleichs zuweist. Gib jede Variable aus.

1.7 Strings konkatenieren

Strings kann man mit einem Plus-Operator aneinander hängen, auch konkatenieren genannt (ersterString + zweiterString + ...). Auch andere Datentypen lassen sich damit in Strings wandeln. Wahrscheinlich hast du genau das schon mal innerhalb von `System.out.println()` benutzt, denn hier passiert nichts anderes als dass alle durch + getrennten Teile in einen String konvertiert werden, hintereinander gehängt und anschließend ausgegeben werden.

Schreibe ein Programm in dem du Variablen aller Typen, die du bisher kennst, anlegst und speichere deren Konkatenation in einem String, den du anschließend aus gibst. (Hinweis: eventuell muss die Konkatenation mindestens einen String enthalten, dies kann aber auch ein Leerer (" ") sein.)

1.8 Byte und Short

Java bietet verschiedene Datentypen für numerische Werte an, die kleinsten Datentypen für ganzzahlige Werte sind `Byte` und `Short`. Erstelle eine Integer-Variable `a` und weise ihr den Wert 127 zu. Deklariere nun eine Variable `b` vom Typ `byte` und eine Variable `c` vom Typ `short`. Weise nun den Variablen `b` und `c` den Wert von `a` zu (beachte hierbei die unterschiedlichen Datentypen). Lasse dir die Variablen nun ausgeben.

Ändere nun dein ursprüngliches Programm so ab, dass die Integer-Variable den Wert 128 besitzt. Was passiert bei der Ausgabe?

2 Bonusaufgaben

In diesem Abschnitt stehen ab sofort Bonusaufgaben. Bonusaufgaben sind optional, du benötigst sie also nicht zum Verständnis. Sie sind da, um mit ihnen bestimmte Teile des Stoffs nochmal üben zu können.

2.1 Erste Logikaufgaben

- Schreibe ein Programm, das einige boolsche Variablen deklariert, ihnen Werte zuweist und diese dann Und-, Oder- und Nicht-verknüpft. Gib die Ergebnisse aus. Versuche ruhig mehrfache Verknüpfungen vorzunehmen.
- Nimm nun einige Integer-Variablen hinzu und verknüpfe Vergleiche (z.B. \leq , $==$) dieser Variablen so, dass eine komplexere Bedingung abgefragt wird. Wozu dient der $!=$ Vergleichsoperator?
- Java nutzt verschiedene Prioritäten, wenn es einen Ausdruck auswertet. Bei $a+b*c$ zum Beispiel wird Punktrechnung vor Strichrechnung ausgeführt. Wie ist das bei boolschen Verknüpfungen? Finde heraus, ob $\&\&$ oder $\|\|$ stärker bindet. Das kannst du zum Beispiel tun, indem du den Ausdruck $a \ \&\& \ b \ \|\| \ c$ nutzt. Wie ist es beim exklusiven Oder (\wedge) und beim Nicht (!)?

2.2 Logische Verknüpfungen für eine Zimmerlampe

In einem Zimmer gibt es eine Lampe und drei Schalter. Gesucht ist nun ein boolscher Ausdruck, so dass die Lampe leuchtet wenn die Mehrheit der drei Schalter gedrückt ist. Schreibe ein Programm, welches für von dir definierte Variablen ausgibt ob die Lampe leuchtet oder nicht.

3 Für zu Hause

Installiere dir das Java Development Kit von `java.sun.com` und probiere das Hello World aus. In der Einleitung im Skript gibt es dazu einen Teil über Java für die verschiedenen Betriebssysteme. Nächste Woche beginnen wir mit Kontrollstrukturen. Lies dazu im Skript das Kapitel *Kontrollstrukturen*.