

Praktikum zu
**Einführung in die Informatik für
LogWings und WiMas**
Wintersemester 2014/15

Übungsblatt 2

Bearbeitungszeit:

3.11.14 - 7.11.14

Aufgabe 2.1 – Zahlensysteme

In dieser Aufgabe sollen Sie Binär- und Dezimaldarstellungen von Zahlen ineinander umrechnen, wie es auch schon in der Vorlesung demonstriert wurde. Fügen Sie in den folgenden Tabellen die fehlenden Werte ein.

Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Dezimalzahl
0	0	1	0	1	5
0	1	1	0	0	
0	0	0	1	0	
1	0	1	1	1	
1	1	0	0	1	

Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Dezimalzahl
1	0	0	0	0	16
					31
					26
					13
					7
					42

Aufgabe 2.2 – Programmieren wie die Großeltern

Für die Bearbeitung dieser Aufgabe sollen Sie *kein* Eclipse starten! Damit wir die Verwendung einer Entwicklungsumgebung richtig zu schätzen lernen, werden wir nun ein Java - Programm ohne Hilfsmittel schreiben.

Bedenken Sie, dass Ihnen während der Klausur ebenso keinerlei Hilfsmittel wie das automatische Prüfen der Java-Syntax oder die Vorgabe von Code-Fragmenten zur Verfügung steht!

- Öffnen Sie einen Texteditor (wir empfehlen `kate` unter KDE) und schreiben Sie das „Hello World“-Programm aus dem ersten Übungsblatt in diese Datei. Übernehmen Sie dabei aber nicht die Paketinformation aus der ersten Zeile, die Ihnen Eclipse vorgegeben hat (z.B. `package blatt01;`). Speichern Sie die Datei als „HelloWorld.java“ ab.
- Öffnen Sie ein Terminal und navigieren Sie zum Verzeichnis, an welchem Sie diese Datei soeben gespeichert haben.
- Führen Sie den Java-Compiler `javac` auf die `.java` Datei aus, indem Sie im Terminal `javac HelloWorld.java` eingeben. Dies sollte nun entweder Fehlermeldungen ausgeben, oder bei Erfolg eine `.class` Datei mit gleichem Namen im Verzeichnis generieren.
- Führen Sie, falls ihnen keine Fehler gemeldet wurden, das Programm aus, indem sie `java` mit dem Programmnamen ausführen: `java HelloWorld`
Häufiger Fehler: Die Dateiendung `.class` darf hier *nicht* mit übergeben werden!
- Wenn Sie „Hello World“ in der Ausgabe sehen, war der Versuch erfolgreich.

Hinweise:

- Sie finden das Terminal-Programm im K-Menü, indem Sie einfach dort danach suchen!
- Im Terminal können Sie mit dem Programm `cd` (*change directory*) navigieren, indem Sie den Pfad relativ zum aktuellen Verzeichnis, oder den absoluten Pfad zum Ziel angeben, z.B. `cd /home/einiXXX/blatt02`, `cd ~/blatt02` oder `cd blatt02`, wenn Sie in ihrem Heimverzeichnis ein Verzeichnis mit dem Namen „blatt02“ angelegt haben.
- Sie können sich den Inhalt des aktuellen Verzeichnisses mit dem Programm `ls` (*list*) anzeigen lassen.
- Ihr Heimverzeichnis ist `/home/einiXXX/`. Sie erreichen dieses jederzeit, indem Sie im Terminal `cd`, oder `cd ~` eingeben.
- Es empfiehlt sich der Einfachheit halber für diesen kurzen Versuch die `.java` Datei daher in ihrem Heimverzeichnis oder in einem Unterordner namens „Blatt2“ zu bearbeiten.

Wir werden im Folgenden ein komplexeres Java-Programm mit Eclipse erstellen.

Aufgabe 2.3 – Vorbereitung des Programmes

Unser Ziel ist es, ein Programm zur Umrechnung von Celsius in Fahrenheit und Fahrenheit in Celsius zu schreiben.

Öffnen Sie Eclipse und erstellen Sie ein neues Paket im `src`-Verzeichnis mit dem Namen „blatt02“ und eine neue Klasse mit dem Namen „Thermometer“.

Eclipse wird Ihnen daraufhin ein — aus Blatt 1 bekanntes — Grundgerüst einer Java Klasse liefern. Vergleichen Sie dieses mit der „Hello World“-Klasse aus Blatt 1 und notieren Sie ihre Vermutung, wie Java erkennt, wo der Programmfluss beginnen soll:

Aufgabe 2.4 – Vorbereitung des Programmierenden

Kein Programmierer fängt an zu arbeiten, ohne sich vorher ausreichend vorbereitet zu haben. Bevor Sie also ein Programm schreiben, sollten Sie sich zuvor darüber Gedanken machen, wie Sie das Problem lösen können, für welches Sie ein Programm schreiben wollen.

Die Formel für die Umrechnung von Celsius in Fahrenheit ist:

$$(\text{Fahrenheit}) = (\text{Celsius}) \cdot \frac{9}{5} + 32$$

Formen Sie zunächst die Formel derart um, sodass Sie eine Formel zur Berechnung von Fahrenheit in Celsius erhalten.

Berechnen Sie anschließend — per Hand — ein paar Beispielwerte und tragen Sie diese erwarteten Ergebnisse in folgender Tabelle ein:

Grad Celsius	Grad Fahrenheit	
	erwartetes Ergebnis	tatsächliches Ergebnis
20		
25		
28		
42		
100		

Aufgabe 2.5 – Variablen und Ausgabe

Spätestens jetzt sollten Sie mit ihrer Praktikumsgruppe die Frage aus Aufgabe 2.3 beantwortet haben. Ergänzen Sie im Programmgerüst die Struktur, die den Start des Programmflusses definiert und fügen Sie dort folgende Codezeilen ein:

```
int fahrenheit;  
int celsius;  
celsius = 20;  
System.out.print("celsius = "+celsius);
```

Bevor Sie das Programm testen, was erwarten Sie wird die 4. Codezeile bewirken?

Fügen sie anschließend folgende Codezeile ein:

```
System.out.println("fahrenheit = "+fahrenheit);
```

Eclipse müsste Ihnen an dieser Stelle eine Fehlermeldung anzeigen. Wie können Sie sich diese erklären?

Kommentieren Sie die 5. Zeile aus, indem Sie ein // vor die Zeile schreiben, und fügen Sie folgende 6. Zeile ein:

```
System.out.println("begin calculation:");
```

Das Auskommentieren einer Zeile sorgt dafür, dass diese nicht mehr vom Compiler berücksichtigt wird. Somit können Sie theoretisch gültige Programmzeilen „archivieren“, um sie später wieder leicht wiederherstellen zu können, indem Sie die Kommentarstriche wieder entfernen. In Kommentaren werden ebenso gerne Erklärungen zum Programm angegeben.

Was fällt Ihnen an der Ausgabe auf?

Modifizieren Sie die 4. Zeile folgendermaßen:

```
System.out.print("celsius = "+celsius+"\n");
```

Fügen Sie anschließend eine 7. Zeile ein:

```
System.out.println("calculation finished");
```

Was fällt Ihnen nun an Zeile 6 im Zusammenhang zur modifizierten 4. Zeile auf?

Diskutieren Sie ihre Ergebnisse mit der Praktikumsgruppe.

Aufgabe 2.6 – Programmieren der Umrechnungsformel

Erweitern Sie das bisher geschriebene Programm, indem Sie zum einen die auskommentierte 4. Zeile zwischen die Ausgabe *begin calculation:* und *calculation finished* verschieben und die Kommentarmarkierung entfernen.

Fügen Sie anschließend vor diese Zeile folgende Anweisung ein:

```
fahrenheit = celsius * 9 / 5 + 32;
```

Testen Sie das Programm und notieren Sie das Ergebnis in der Tabelle von Aufgabe 2.4.

Ändern Sie anschließend das Programm, um die anderen Daten aus der Tabelle zu berechnen.

Was fällt Ihnen am Ergebnis des Programms auf?

Diskutieren Sie das Ergebnis in der Praktikumsgruppe unter dem Aspekt der Java Datentypen, die Sie aus der Vorlesung kennen.

Aufgabe 2.7 – Die Java Language Specification

In Kapitel 1 (Folien 10–33) der Vorlesung haben Sie unterschiedliche Datendarstellungen kennengelernt. Schauen Sie sich diese Folien noch einmal an und betrachten Sie zusätzlich den folgenden Text auf der Oracle-Webseite.

<http://docs.oracle.com/javase/specs/jls/se7/jls7.pdf>

Lesen Sie den Abschnitt 4.2 über primitive Datentypen. Tragen Sie dann alle primitiven Datentypen der Programmiersprache Java in der folgenden Tabelle ein und beschreiben Sie kurz in eigenen Worten, worum es sich bei dem jeweiligen Datentyp handelt.

Datentyp	Beschreibung

Hinweis: Wenn Sie ein tieferes Verständnis für die Programmiersprache Java entwickeln wollen, lohnt es sich das oben verlinkte Dokument genauer durchzulesen — dies ist allerdings weder für die Vorlesung noch für das Praktikum notwendig.

Lesen Sie anschließend zusätzlich die Abschnitte 4.3.3 und 3.10.5 über Strings und notieren Sie, was Ihnen wichtig erscheint!

Diskutieren Sie anschließend die Ergebnisse mit der Praktikumsgruppe und erarbeiten Sie gemeinsam eine Lösung für die falschen Ergebnisse in Aufgabe 2.3.

Aufgabe 2.8 – Rückrichtung

Schreiben Sie ihren Code so um, dass Sie nun zuerst die Fahrenheit im Quellcode angeben und die Temperatur in Celsius umrechnen. Ändern Sie entsprechend auch die Ausgaben des Programmes.