

Informatik I WS 06/07
Prof. Dr. C. Damm
Dipl.-Inform. Marc Njoku

Übungsblatt 4

Ausgegeben am:	15.11.2006
Abgabe bis:	24.11.2006

Thema: Wahrheitstafeln, Algorithmen, Java

Die Theorieaufgaben auf diesem Blatt sind bis zum Freitag, 24. November um 14.00 Uhr in die Info-I-Briefkästen im Erdgeschoss der NAM einzuwerfen. Die Kästen sind nach Übungsgruppen geordnet. Achten Sie bitte darauf dass Sie Ihren Zettel in den richtigen Kasten werfen, und dass **Name und Gruppe** auf **jedem Blatt** stehen! Falls Ihre Lösung mehrere Blätter umfasst, heften Sie diese bitte zusammen. Die Praxis-Aufgaben bearbeiten Sie bitte am Rechner und führen Sie Ihrem Tutor am Rechner vor. Eine Bearbeitung in Zweier-Teams innerhalb Ihrer Übungsgruppe ist möglich.

Aufgabe 1 (Theorie: 16 Punkte):

Wahrheitstafeln

Geben Sie zu folgenden aussagenlogischen Formeln die Wahrheitstafeln an:

- $F_1 \equiv \neg A$
- $F_2 \equiv (\neg A) \wedge B$
- $F_3 \equiv (\neg A) \vee B$
- $F_4 \equiv A \wedge ((\neg B) \vee C)$
- $F_5 \equiv (A \wedge (\neg B)) \vee C$
- $F_6 \equiv \neg(\neg A \vee B) \wedge \neg(A \vee (\neg B))$
- $F_7 \equiv A$

Aufgabe 2 (Theorie: 16 Punkte):

Pseudocode und Anweisungen

Gegeben seien folgende Pseudocode-Ausdrücke:

- Bedingte Verzweigung: **if** B **then** A_1 **else** A_2 **fi**
Wenn Bedingung B gilt, wird Anweisung A_1 ausgeführt, sonst Anweisung A_2 . Das Schlüsselwort **fi** zeigt das Ende der bedingten Verzweigung an.
- Abweisende Schleife: **while** B **do** A **od**
Solange Bedingung B gilt wird Anweisung A ausgeführt. Das Schlüsselwort **od** zeigt das Ende der Schleife an.
- Zuweisung: $X := E$
Der Variablen X wird der Wert des Ausdrucks E zugewiesen.
- Logische Konnektive: **and**, **or**, **not**
Die Konnektive für Konjunktion, Disjunktion und Negation.
- Wahrheitswerte: **true**, **false**
Die Wahrheitswerte für wahr und falsch.

Zeigen Sie unter Angabe einer Folge von Anweisungen an, die in ihrer Gesamtheit der bedingten Verzweigung **if** B **then** A_1 **else** A_2 **fi** entsprechen, dass sich die besagte Verzweigung mit Hilfe von

abweisenden Schleifen umsetzen lässt (ohne die bedingte Verzweigung selbst zu benutzen!).

Aufgabe 3 (Theorie: 24 Punkte):

Java

Die von der Firma SUN entwickelte Sprache Java ist angeblich

- einfach & klein
- objektorientiert & verteilt
- interpretiert & schnell
- robust & sicher
- plattformunabhängig & portierbar
- multithread-fähig & dynamisch

Was bedeuten diese Schlagwörter? Diskutieren Sie diese Eigenschaften.

Aufgabe 4 (Praktisch: 14 Punkte):

Java

Der Compiler, der aus einem Quelltext ausführbaren Programmcode herstellt, heißt bei der hier verwendeten Java-Variante von Sun **javac**. Er wird normalerweise mit dem Namen der zu kompilierenden Klassendatei und eventuell zusätzlichen Parametern aufgerufen, z.B. **javac Info1Class.java**

Tippe zunächst einmal das folgende kurze Programm ab. Es soll Dein erstes Java-Programm werden. Speichere es als *HalloWelt.java* ab (beachte dabei die Groß- und Kleinschreibung). Kompiliere das Programm mit dem Befehl '**javac HalloWelt.java**' und führe es mit dem Befehl '**java HalloWelt**' aus.

```
class HalloWelt
{
    static public void main(String[] args)
    {
        System.out.println("Hallo Welt!");
    }
}
```

Ist ein Quelltext syntaktisch oder grammatikalisch fehlerhaft, so kann schon der Compiler Fehler finden. Das hilft Ihnen eventuell, wenn Sie beim Abtippen des obigen Programmes Fehler gemacht haben.

Beim Aufruf eines kompiliertes Java-Programms ist es möglich, Parameter mit zu übergeben. Die übergebenen Parameter sind über das Zeichenkettenfeld *args* aus dem Programm heraus zugreifbar. Die Nummerierung erfolgt wie in der Informatik üblich bei 0 beginnend.

Ändern Sie nun das Programm *HalloWelt* so ab, dass man es mit

```
java HalloWelt Dein Name
```

aufrufen kann und es dann statt "*Hallo Welt!*" den Gruß "*Hallo Dein Name!*" ausgibt.

Aufgabe 5 (Praktisch: 30 Punkte):

Java

Tippen Sie das folgende, aus zwei Dateien bestehende Programm ein und lassen Sie es übersetzen.

Datei Aufgabe4_5.java:

```
class Aufgabe4_5
{
    static public void main(String[] args)
    {
        Hallo hallo = Hallo("Java ist ...");
        hallo.ausgabe();
    }
}
```

Datei Hallo.java:

```
import java.util.*;
import java.io.*;

class Hallo
{
    String text;

    Hallo(string text)
    {
        this.text = "Heute ist " + Calendar.getInstance().getTime() +
            " und " + text;
    }

    void ausgabe()
    {
        System.out.println(text);
    }
}
```

Hierzu können sie ein Verzeichnis namens '*Aufgabe4_5*' erzeugen. Beim Übersetzen (jeweils mit dem Befehl '**javac** *Name_der_Javadataei*') werden Sie zwei Fehlermeldungen erhalten, die Sie auch ohne Java-Kenntnisse korrigieren können. Ersetzen Sie die Zeichenfolge "..." durch eine angemessene Aussage und starten (mit dem Befehl '**java** *Name_der_Klasse*') Sie das Programm.

Wenn das Programm läuft und Sie anschließend die Zeichenfolge "..." geeignet ersetzt haben, übersetzen und starten Sie es wenigstens einmal auch mit direkten Kommandos (**javac**, **java**).