Institut für Informatik
der Universität München
Prof. Dr. M. Hofmann
Dr. R. Matthes, M. Lange

WS 02/03

16. Oktober 2002

Practical 0 zur Vorlesung

Einführung in die Informatik: Programmierung und Software-Technik

Abgabe: spätestens 30.10.2002, 14:00 Uhr

Ziel dieses Practicals ist es, Sie mit dem bekannt zu machen, was Sie für die folgenden Practicals brauchen. Dazu gehört neben dem generellen Umgang mit Computern, dem Internet, usw. vor allem das Editieren von Dateien, die Ihre Programme beinhalten werden, sowie das Benutzen des Java-Compilers.

Falls Sie mit einigen oder allen Punkten dieses Practicals bereits vertraut sind, dann überspringen Sie sie einfach. Dieses Practical ist unbewertet, es gibt keine Punkte. Dennoch müssen Sie im Rahmen dieses Practicals etwas abgeben.

Bei Problemen wenden Sie sich an die studentischen Hilfskräfte, die die Rechnerraumbetreuung durchführen. Diese sind ab Montag, 21.10., an den folgende Terminen im CIP-Pool Sibirien bzw. Gobi zu finden.

Montag	9:00	–	13:00	Schmarsow
Montag	14:00	–	16:00	Domin
Montag	16:00	–	18:00	Dax
Montag	18:00	–	20:00	Domin
Dienstag	9:00	–	13:00	Khoma
Dienstag	14:30	–	18:30	Reitmeier
Mittwoch	9:00	–	13:00	Haddad
Donnerstag	12:00	–	14:00	Dax
Freitag	11:30	–	15:30	Wu

1 Computer, Linux, KDE, ...

1. Um die Computer in den CIP-Pools im Keller der Oettingenstrasse 67 bzw. in den anliegenden Barackengebäuden benutzen zu können, brauchen Sie eine Kennung. Diese erhalten Sie bei der Rechnerbetriebsgruppe Informatik an den folgenden Terminen, abhängig vom Anfangsbuchstaben Ihres Nachnamen.

Montag	A – H	(14. Oktober)
Dienstag	I – S	
Mittwoch	S – Z	
Donnerstag	A – P	
Freitag	Q – Z	(18. Oktober)
Montag	A – M	(21. Oktober)
Dienstag	N – Z	(22. Oktober)

jeweils zwischen 18 und 19 Uhr im Raum 003/Arktis (weisser Flachbau im Norden des Geländes der Oettingenstr. 67). Benötigt wird der Studentenausweis, ein amtl. Ausweis mit Lichtbild (damit ist ein Personalausweis, Führerschein, etc. gemeint, der ein Bild beinhaltet und nicht dass ein Lichbild mitgebracht werden muss!) sowie die persönliche Anwesenheit, da eine Unterschrift geleistet werden muss.

2. Finden Sie einen freien Computer und loggen Sie sich ein. Zuerst geben Sie Ihr Kennwort ein, danach Ihr Passwort. Kurze Zeit später sehen Sie die KDE–Oberfläche. Hier finden Sie alles, was sie brauchen.

3. Normalerweise sollte sich bereits ein Webbrowser geöffnet haben. Falls nicht, dann benutzen Sie das Zahnrad in der unteren linken Ecke, um ein Menü zu öffnen, in dem Sie einen Webbrowser suchen können.

4. KDE unterstützt virtuelle Bildschirme (*Desktops*). Sie müssen also nicht jedes Programm, das läuft, ständig vor sich haben. Stattdessen können Sie es auf einen dieser virtuellen Bildschirme legen. Klicken Sie dazu die linke obere Ecke des entsprechenden Fensters an, indem Sie den Mauszeiger dorthin bewegen und die linke Maustaste kurz drücken. Ein Menü öffnet sich. Suchen Sie den Menüpunkt, mit dem sich das Fenster auf einen anderen Desktop legen läßt. Tun Sie dies. Die virtuellen Bildschirme werden in der unteren Bildleiste in der Mitte angezeigt. Gewechselt wird durch Klicken auf diese. Versuchen Sie, Ihr Fenster wiederzufinden.

5. Zum Editieren von Dateien brauchen Sie einen Editor (wie der Name es schon sagt . . .). Versuchen Sie, im Menü unten links einen Eintrag “Emacs” bzw. “XEmacs” in einem der Untermenüs zu finden und Emacs zu starten.

   Emacs kann vieles und noch viel mehr. Die wichtigsten Funktionen wie Öffnen und Schließen von Dateien, Speichern und Beenden sind über die Menüs am oberen Rand des Fensters zu erreichen.

2 Dateien, Verzeichnisse, . . .

1. Starten Sie eine *Terminal-Emulation*. In der unteren Bildleiste sehen Sie das Icon eines Bildschirms hinter einer Muschel. Darauf klicken! Über die Shell können Sie mit dem Computer (genauer gesagt mit dem Betriebssystem Linux) kommunizieren.

2. Ziehen Sie das Fenster mit der Maus an ihre Lieblingsposition. (nicht klausurrelevant)

3. Mit Ihrer Kennung erhalten Sie Speicherplatz auf den Rechnern der CIP-Pools, Ihr sogenanntes *Homeverzeichnis*. Dies ist unabhängig von dem Rechner, an dem Sie sitzen, Sie können also beim nächsten Einloggen einen anderen Rechner nehmen.

4. Lassen Sie sich den Inhalt Ihres Homeverzeichnisses anzeigen.

   `ls`

5. Erstellen Sie ein Unterverzeichnis. Denken Sie sich dazu einen besonders tollen Namen aus.

   `mkdir` *Name*

6. Wechseln Sie von Ihrem Homeverzeichnis in das neue Unterverzeichnis.

   `cd` *Name*

   Lassen Sie sich dort den Inhalt anzeigen. Nichts da? Kein Wunder.

7. Erstellen Sie dort eine Datei. Denken Sie sich dazu einen zweiten Namen aus.

   `touch` *Name2*

   Dies werden Sie später kaum brauchen, denn jedes normale Anwendungsprogramm erstellt Dateien für Sie. Lassen Sie sich anzeigen, wann die Datei erstellt wurde und wir groß sie ist.

   `ls -al` *Name2*

   Oftmals wird der Sinn bzw. Typ einer Datei durch eine Endung deutlich gemacht. Löschen Sie die bereits erstellte Datei wieder.

   `rm` *Name2*

   Erstellen Sie *Name2*`.java`.

8. Wechseln Sie in das Verzeichnis über dem aktuellen.

   `cd ..`

9. Kopieren Sie die Datei aus dem Unterverzeichnis in das aktuelle. Dies wird immer durch einen Punkt gekennzeichnet, während zwei Punkte das darüberliegende bezeichnet.

   `cp` *Name*/*Name2*`.java .`

10. Löschen Sie das Unterverzeichnis.

    `rmdir` *Name*

    Geht nicht. Zuerst müssen Sie Ihre Datei aus dem Unterverzeichnis entfernen. Entweder Sie löschen sie, oder Sie verschieben sie in ein anderes Verzeichnis, z.B. das darüberliegende.

```
cd Name
mv Name2.java ..
```

Beachten Sie, dass dadurch die gleichnamige Datei im Verzeichnis `..` überschrieben wurde. Außerdem läßt sich `mv` auch zum Umbenennen von Dateien benutzen. Gehen Sie in das Verzeichnis, in dem *Name2*`.java` liegt, und denken Sie sich einen weiteren Namen aus.

```
mv Name2.java Name3
```

Lassen Sie sich wieder den Inhalt des Verzeichnisses anzeigen.

3 Internet, WWW, Forum

1. Starten Sie einen WWW-Browser wie z.B. Mozilla, Netscape Navigator bzw. Communicator oder Konqueror. Diese finden Sie entweder im Menü unten links oder als Icons auf Ihrem Desktop.

   Es gibt prinzipiell zwei Möglichkeiten, auf eine Seite im WWW zuzugreifen und sich diese anzeigen zu lassen. Entweder Sie geben die Adresse der Seite (URL) dem Browser explizit an, indem Sie sie in das URL-Feld eintippen. Tun Sie dies für

   `http://www.tcs.informatik.uni-muenchen.de/lehre/WS02-03/PSE/`

   Oder Sie klicken sich sukzessive entlang von *Links* durch das WWW. Diese sind normalerweise blau auf einer WWW-Seite dargestellt und eventuell unterstrichen. Durch Anklicken gelangen Sie zu der Seite, auf die der Link verweist. Versuchen Sie so, von der WWW-Seite der Vorlesung aus, sich das Java-Tutorial anzeigen zu lassen.

2. Auf der WWW-Seite der Vorlesung liegt ein Link zum Forum, in dem allgemeine Fragen zur Vorlesung, den Übungen, usw., die alle Hörer der Vorlesung betreffen, gestellt werden und hoffentlich auch beantwortet werden können. Fragen werden per Mail an

   `einfinfo@tcs.informatik.uni-muenchen.de`

   in das Forum eingetragen. Die URL des Mailarchivs ist

   `http://www.tcs.informatik.uni-muenchen.de/lehre/WS02-03/EinfInfoListe/`

   Schauen Sie sich das Forum zumindest einmal an.

4 Emacs

1. Öffnen Sie eine Shell und legen Sie dort ein Unterverzeichnis mit einem beliebigen Namen an.

2. Starten Sie Emacs oder XEmacs, von nun an nur noch Emacs genannt.

3. Emacs unterstützt das gleichzeitige Bearbeiten mehrerer Dateien in sogenannten *Buffern*. Emacs startet mit dem Buffer `*scratch*`. Diesen sollten Sie nicht zum Editieren von Dateien benutzen. Öffnen Sie stattdessen einen neuen über den Menüpunkt `File -> Open`. Sie erhalten ein Fenster, in dem Sie den Namen des Buffers, der beim Abspeichern auch als Dateiname dienen wird, und ein existierendes Verzeichnis auswählen können. Tun Sie dies mit einem beliebigen Namen und dem gerade angelegten Verzeichnis.

4. Geben Sie eine Zeile Text in Ihren Buffer ein und speichern Sie diese über den Menüpunkt `File -> Save` ab.

5. Wechseln Sie in der Shell in das Verzeichnis, in dem Emacs Ihre Datei gespeichert hat.

6. Lassen Sie sich dort den Inhalt Ihrer Datei anzeigen.

   `cat` *`Dateiname`*

7. Beenden Sie Emacs.

5 Java

1. Auf der WWW-Seite der Vorlesung liegt ein Link zu einem kleinen Java-Programm mit dem Namen `Versuch.java`. Um dies auszuführen muß es zuerst kompiliert, d.h. in Maschinencode übersetzt werden. Dazu müssen Sie es jedoch vorher auf Ihrem Rechner haben. Legen Sie dazu wieder ein geeignetes Unterverzeichnis an.

   Suchen Sie den Link zu `Versuch.java`.

2. Sie haben mehrere Möglichkeiten, `Versuch.java` auf Ihren Rechner zu holen. Alle Browser unterstützen das Abspeichern von Dateien, auf die ein Link verweist. Dies geschieht normalerweise dadurch, dass Sie den Mauszeiger auf den Link bewegen und die rechte Maustaste drücken. Ein Menü öffnet sich. Wählen Sie den Menüpunkt zum Abspeichern (z.B. `Save as ...`), und speichern Sie das Beispielprogramm unter dem Namen `Versuch.java` in dem angelegten Unterverzeichnis.

3. Lassen Sie sich den Inhalt der gespeicherten Datei anzeigen.

4. Eine andere Möglichkeit besteht darin, den Link auf `Versuch.java` im WWW-Browser zu öffnen. Da es sich um eine Text-Datei handelt, sollte Ihr Browser diese anzeigen können. Öffnen Sie einen Buffer `Versuch.java` im Emacs für das angelegte Unterverzeichnis. Markieren Sie das Java-Programm im WWW-Browser, indem Sie den Mauszeiger bei gedrückter, linker Maustaste darüberziehen. Wechseln Sie zum Emacs–Buffer `Versuch.java` und klicken Sie mit der mittleren Maustaste in das Textfeld. Das Programm sollte dort erscheinen. Speichern Sie es ab.

5. Für diejenigen, denen diese zwei Möglichkeiten noch nicht reichen: Tippen Sie das Programm ab.

6. Nun kann das Programm kompiliert werden. Öffnen Sie dazu eine Shell und wechseln Sie in das entsprechende Unterverzeichnis.

7. Testen Sie, ob der Java-Compiler verfügbar ist.

   ```
   which javac
   ```

8. Kompilieren Sie das Programm.

   ```
   javac Versuch.java
   ```

   Welche Dateien wurden erzeugt? Falls sich Emacs im Java-Modus befunden hat und in der Menüleiste der Punkt JDE sichtbar ist, so können Sie das Programm auch durch Anklicken des Unterpunktes `Compile` kompilieren.

9. Das Programm sollte sich jetzt ausführen lassen. Dazu brauchen Sie den Java-Interpreter `java`. Testen Sie, ob dieser verfügbar ist.

10. Führen Sie das Programm aus.

    ```
    java Versuch
    ```

11. Kommentieren Sie nun die erste Zeile nach der `for`-Anweisung aus.

    ```
    /* if (( ...)) { */
    ```

12. Speichern Sie den Buffer ab und kompilieren Sie das Programm erneut. Sie erhalten eine oder mehrere Fehlermeldungen. Kein Wunder, schließlich paßt zu der letzten schließenden Klammer jetzt keine öffnende mehr, und dadurch erkennt der Compiler das Programm nicht mehr als gültig an.

13. Fügen Sie gleich nach der auskommentierten Zeile die folgende ein.

    ```
    if ((i % 2 == 0) && ((i == 30) || (i % 6 != 0)) && (i != 14)) {
    ```

14. Speichern Sie das Programm wieder ab, kompilieren Sie es erneut und führen Sie es aus.

15. Zum Schluss wird das Programm noch komprimiert und verpackt, bevor Sie es probeweise mit unserem elektronischen Abgabesystem abgeben. Verpacken spart zwar bei kleinen Programmen und nur einer Datei nicht viel ein. Haben Sie jedoch mehrere Dateien, was bei größeren Java-Programmen oft unvermeidlich ist, dann können Sie diese so alle zusammen in einer Datei archivieren, um sich dann nur noch um eine einzige Datei kümmern zu müssen.

    Das Programm zum Verpacken von Java-Dateien heißt `jar`. Testen Sie, ob es verfügbar ist.

    Denken Sie sich einen Namen für Ihr Archiv aus und geben Sie folgendes ein.

    ```
    jar cvf Name.jar Versuch.java
    ```

`jar` erzeugt eine Statusmeldung, aus der ersichtlich wird, inwieweit sich das Programm komprimieren ließ. Ausserdem wurde das Archiv *Name*.`jar` als Datei erzeugt. Zur Erläuterung: Der Parameter `c` steht dafür, ein Archiv anzulegen, `v` besagt, dass `jar` Meldungen ausgeben soll, und `f` zeigt `jar` an, dass die folgende Zeichenkette der Name des anzulegenden Archivs sein soll. Hinter diesem stehen dann alle Dateien, die in das Archiv gepackt werden sollen.

6 Elektronisches Abgabesystem

1. Zuerst einmal müssen Sie sich anmelden. Gehen Sie dazu mit Ihrem Webbrowser auf die WWW-Seite der Vorlesung und klicken Sie sich von dort zur Anmeldeseite des Abgabesystems. Beachten Sie, dass Sie sich bis zum 30.10., 14:00 Uhr, angemeldet haben müssen.

2. Sie erhalten ein Formular, das Sie auffordert, Ihren Vornamen, Namen, Ihre Matrikelnummer, Ihr Geschlecht und Ihre Emailadresse anzugeben. Wenn Sie eine Kennung am CIP-Pool haben, haben Sie auch eine Email-Adresse. Diese lautet

   kennung`@informatik.uni-muenchen.de`

   Falls Sie bereits eine andere Email-Adresse haben, können Sie diese natürlich benutzen.

   Tragen Sie die Daten ein und schicken Sie das Formular ab. Sie erhalten eine Bestätigung. Von da an werden Sie über Ihre Matrikelnummer identifiziert. Haben Sie sich bei der Matrikelnummer vertippt, dann melden Sie sich einfach nochmal an. Haben Sie sich bei Ihrem Namen vertan, dann kontaktieren Sie uns per Email, da eine zweite Anmeldung mit der gleichen Matrikelnummer nicht möglich ist.

3. Gehen Sie nun über die Vorlesungs-WWW-Seite zur Abgabeseite des Systems. Dort erwartet Sie ebenfalls ein Formular. Für die folgenden Practicals ist Teamarbeit und Abgabe in Gruppen zu jeweils drei Personen empfohlen und gewünscht. Dann werden Sie dort die Möglichkeit haben, insgesamt drei Matrikelnummern einzutragen. Für dieses Practical gilt jedoch, dass Sie alleine abgeben müssen. Tragen Sie Ihre Matrikelnummer in das dafür vorgesehene Feld ein.

   Dann müssen Sie Ihrem Webbrowser noch mitteilen, welche Datei er für Sie an das Abgabesystem übertragen soll, sprich wo sich das von Ihnen angelegte Java-Archiv *Name*.`jar` befindet. Wenn Sie das genaue Verzeichnis kennen, können Sie es in das entsprechende Feld per Hand eintragen. Oder Sie klicken auf den `Browse`-Button. Es öffnet sich ein Dialogfenster, mithilfe dessen Sie das Archiv lokalisieren können, indem Sie sich durch die einzelnen Verzeichnisse bis in das entsprechende durchklicken. Achten Sie darauf, dass der *Filter* in diesem Fenster auf etwas wie `*` oder `*.jar` gesetzt ist, da ansonsten Ihre Datei eventuell nicht angezeigt wird.

   Damit das elektronische Abgabesystem weiß, was es zu kompilieren hat, wird bei jeder Abgabe ein Klassen– und Dateiname vorgegeben. Fuer dieses Practical ist dies

```
Versuch
```

Das bedeutet, dass in Ihrem `jar`–Archiv eine Datei mit dem Namen `Versuch.java` vorhanden sein muß. Desweiteren muß diese Datei eine Klasse mit dem Name `Versuch` beinhalten. Dies ist durch die Zeile

```
public class Versuch {
```

gewährleistet. (Das Programm wird nicht kompilieren, wenn Sie nur den Namen der Datei ändern! Tun Sie dies zur Übung und versuchen Sie es zu kompilieren.)

Wenn Sie Ihr Archiv gefunden haben, klicken Sie auf OK, damit sich das Dialogfenster schließt. Dann nur noch auf Abgeben klicken. Warten Sie einen Augenblick. Das Abgabesystem kompiliert Ihr Programm aus dem Archiv heraus. Wenn dies erfolgreich geschehen ist, dann erhalten Sie eine entsprechende Meldung. Anderenfalls erhalten Sie eine Fehlermeldung und das Abgabesystem verweigert die Annahme.

Um dieses zu testen raten wir Ihnen, Ihr Java-Programm an einer beliebigen Stelle zu verändern, so dass es nicht mehr kompiliert. (Ganz sicherer Tip: eine Klammer löschen oder hinzufügen.) Packen Sie es in ein Java-Archiv und geben Sie es über das Abgabesystem ab, um die Fehlermeldung zu bestaunen.

Herzlichen Glückwunsch. Somit haben Sie jetzt das Rüstzeug, um die folgenden Practicals zu meistern. Außerdem können Sie sich dann nicht mit "Ich habe nicht gewußt, wie das geht." herausreden ...