Andreas Podelski
Fachbereich 14 – Informatik
Universität des Saarlandes,
Max-Planck-Institut für Informatik

Programmiersprachen (WS 1996/97)

24. Oktober 1996

Übungsblatt 1

Abdelwaheb Ayari
Hubert Baumeister

Aufgabe 1: (2 Punkte)
Bilde für die folgenden 4 ML-Ausdrücke `E` jeweils einen ML-Ausdruck `P`, wobei der Ausdruck `P` die Variable `x` enthält, so daß bei Anfragen der Art `val P = E` die Variable `x` an `0` gebunden wird.

1. `{eins=1, c=0,a=true}`
2. `[ ~2, ~1, 1, 0 , 2 ]`
3. `[ (1,0), (2,1) ]`
4. `( true, (false, 42 mod 6) , "Hi", 42 div 6 )`

Aufgabe 2: (2 Punkte)
Definiere mit Hilfe von Patternmatching eine Funktion `ifthenelse(A,B,C)`, die sich verhält wie `if A then B else C`. Ist dies möglich? Begründe die Antwort.

Aufgabe 3: (3 Punkte)
Schreibe ein ML-Programm welche folgende Funktionen bereitstellt. Spezifiziere die Funktionen und begründe warum das Programm der Spezifikation genügt.

1. `insert_nth`: Fügt ein Element in eine Liste an n-ter Stelle ein.
2. `delete`: Löscht ein Element aus einer Liste falls vorhanden (eindeutig?).
3. `append`: hängt zwei Listen zusammen.
4. `reverse`: dreht eine gegebene Liste um.
   (Hinweis: benutze `append`. Geht es auch effizienter?)

Aufgabe 4: (3 Punkte)
Implementiere eine der Listenfunktionen von Aufgabe 3 in C oder Pascal und begründe warum das Programm der Spezifikation genügt.

1 Hinweise zum Übungsbetrieb

- Die Abgabe der Lösungen erfolgt in der Regel jeweils am Donnerstag in der Woche nach Ausgabe des Übungsblattes. Für das erste Übungsblatt ist das Donnerstag der 31. 10.

- Gruppenarbeit ist erlaubt. Maximal 3 Personen dürfen zusammen ein Übungsblatt bearbeiten. Es braucht nur eine Lösung pro Gruppe abgegeben werden, auch wenn die einzelnen Gruppenmitglieder in unterschiedliche Übungsgruppen gehen. Allerdings ist es empfehlenswert wenn alle Mitglieder einer Gruppe dieselbe Übungsgruppe besuchen, da spezielle Kommentare zu den Abgaben einer Gruppe in der Regel nur in einer Übungsgruppe gemacht werden. Auf der Abgabe müssen alle Namen der Gruppenmitglieder vermerkt sein.

- Einige Aufgaben auf den Übungsblättern sind Programmieraufgaben. Die Programme **müssen** auf den Rechnern des Cip-Pools laufen. Es ist also nur möglich Programme z.B. zu Hause auf dem PC zu entwickeln, wenn sie vor der Abgabe auf die Rechner des Cip-Pools portiert wurden. Programme müssen als Ausdrucke **und** in elektronischer Form abgegeben werden. Die Programme müssen kommentiert sein und die Ausdrucke **Testläufe** enthalten.

  Die elektronische Abgabe erfolgt per ftp in das Verzeichnis `/pub/incoming/PS` auf dem Rechner `ftp.mpi-sb.mpg.de` unter dem Namen `<Vorname>.<Nachname>.ueb<nr>.aufg<nr>`. Bei Gruppenabgabe reicht ein File wobei alle Teilnehmer als Kommentar am Anfang der Datei aufgeführt werden.

  Beispiel: Susi Sinnlos gibt zum dritten Übungsblatt zusammen mit Rudi Ratlos die 4. Aufgabe ab. Das Programm steht in der Datei `prog.sml` im aktuellen Verzeichnis und die erste Zeile enthält den Kommentar `(* Susi Sinnlos, Rudi Ratlos *)`.

```
susi@cip51(3)% ftp ftp.mpi-sb.mpg.de
Connected to ftp.
220 newftp FTP server (Version wu-2.4(9) Thu Jul 11 17:10:55 MET DST 1996) ready.
Name (ftp.mpi-sb.mpg.de:hubert): ftp
331 Guest login ok, send your complete e-mail address as password.
Your e-mail-Address: susi
[..]
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /pub/incoming/PS
ftp> put prog.sml Susi.Sinnlos.ueb3.aufg4
ftp> quit
221 Goodbye.
```

- Bedingungen für die Scheinvergabe sind 50% der maximalen Punktzahl bei der Bearbeitung der Übungen **und** aktive Teilnahme an den Übungen (z.B. vorrechnen von Übungsaufgaben an der Tafel).

- Alle diese Informationen finden sich auch auf der WWW Seite zur Vorlesung.:
  `http://www.mpi-sb.mpg.de/~hubert/PS96-97/`

2 Standard ML (`sml`)

2.1 sml-Pfad

Der Standard ML Compiler (sml) befindet sich unter `/usr/local/bin/sml`.

2.2 Von der Shell

```
cip48 [15:52] sml
Standard ML of New Jersey, Version 0.93, February 15, 1993
val it = () : unit
-
```

Beenden von `sml` durch Eingabe von C-d (Control und d).

Laden einer Datei `x.sml`

```
- use("x.sml");
[opening x.sml]
val it = () : unit
-
```

2.3 im Emacs

- Kopiere `/home/stud/abdelwaheb/.emacs` nach `$HOME/.emacs`.
- starte Emacs mit `emacs &`.
- Lade eine Datei mit Endung `.sml` → `sml-mode`.
- `C-c C-b` lädt Datei in sml und startet sml wenn notwendig.
- `C-d` im sml-Fenster beendet sml.

Weitere Information über den `sml-mode` mit `C-h m`.

SML-Anleitung im Emacs

- Starten des Informationssystems mit `C-h i`.
- Lernen des Informationssystems *Info* mit `h`.
- SML-Anleitung mit `g`, dann[1] `(/home/stud/abdelwaheb/info/sml)`.

3 WWW

Informationen zur Vorlesung

`http://www.mpi-sb.mpg.de/~hubert/PS96-97`

WWW-Browser: `/usr/local/bin/netscape`, `/usr/local/bin/Mosaic`, `/usr/bin/X11/chimera`
Emacs: `ESC-x w3`

[1] Die Klammern sind nötig!