

Übungen 8: Ein- und Ausgabe

Programmiertechniken in der Computerlinguistik I · Wintersemester 2004/2005

1. Ewige Wahrheiten

Was ist der entscheidende Unterschied zwischen den Prädikaten `true` und `repeat`?

2. Leerzeichen

Das vordefinierte Prädikat `tab/1` schreibt soviele Leerschläge heraus, wie sein Argument enthält. Zeige, dass dieses Prädikat leicht selbst definiert werden kann, indem du ein äquivalentes `my_tab/1` implementierst.

3. Gross- und Kleinschreibung

Trotz amtlicher Rechtschreibereform fürs Deutsche trifft man in weniger amtlichen Bereichen – wie etwa in Emails – auf ganz unterschiedliche Schreibungen. Einige schreiben alles klein, einige nur den Satzanfang gross, einige wie im Englischen Eigennamen gross. Daneben gibt's Tippfehler, oder Grossschreibung wird zum Ausdruck von Emotionen (SCHREIEN) verwendet.

a) Definiere die Prädikate `uc/2` und `lc/2`, die genau dann wahr sein sollen, wenn das 2. Argument die gross- bzw. kleingeschriebene Variante des 1. Arguments ist.

```
?- uc(sbb, X).  
X = 'SBB'  
yes
```

```
?- lc('Wenn hinter Fliegen Fliegen fliegen.', X).  
X = 'wenn hinter fliegen fliegen fliegen.'  
yes
```

Die Argumente sollen Atome sein und die Prädikate müssen nur im Modus `uc(+Atom,?Atom)` bzw. `lc(+Atom, ?Atom)` funktionieren. Die deutschen Umlaute Ä,Ö,Ü,ä,ö,ü sollen ebenfalls berücksichtigt werden.

Tipp: Mit einem geeigneten Hilfsprädikat, das die Beziehung zwischen gross- und kleingeschriebenen Varianten der Buchstaben ausdrückt, wird der Schreibaufwand in dieser Übung markant verringert. Denke daran, dass in ASCII und ISO-Latin 8859-1 die Codes von Gross- und Kleinbuchstaben systematisch miteinander verbunden sind!

c) Definiere das Prädikat `lc_file(+InputDatei,+OutputDatei)`, das alle Prologterme aus `InputDatei` einliest und die kleingeschriebene Variante dieser Terme in `OutputDatei` hineinschreibt. `OutputDatei` wird also erzeugt!

Beispiel: In der Datei "bsp.txt" auf Laufwerk C befindet sich der Text der linken Spalte von untenstehender Tabelle. Nach dem Aufruf von

```
?- lc_file('C:/bsp.txt', 'C:/lcbsp.txt').  
yes
```

enthält die Datei "lcbsp.txt" auf Laufwerk C den Text der rechten Spalte von untenstehender Tabelle. (Beachte die Verzeichnistrenner im UNIX-Stil mit Schrägstrich "/" statt *Backslash* "\")

Inhalt von Datei 'C:/bsp.txt'	Inhalt von Datei 'C:/lcbsp.txt'
'Wenn'.	wenn.
hinter.	hinter.
'Fliegen'.	fliegen.
'Fliegen'.	fliegen.
fliegen.	fliegen.
','.	','.
fliegen.	fliegen.
'Fliegen'.	fliegen.
'Fliegen'.	fliegen.
nach.	nach.
'.'.	'.'.

Tipp: Verwende `write_canonical/1` zum Herausschreiben der Prologterme, damit das Komma und der wörtliche Punkt korrekt zitiert werden.

```
?- write('.,').  
.   
yes  
?- write_canonical('.,').  
'.,'  
yes
```

Wer mit *SICStus Prolog* unter *MacOS X* arbeitet, muss die Dateinamen, die sich nicht auf das momentan aktuelle Arbeitsverzeichnis beziehen im UNIX-Stil angeben. D.H. aus der Datei "bsp.txt" auf dem Volumen "MacHD" die Datei "lcbsp.txt" zu erzeugen, muss der Aufruf

```
?- lc_file('/Volumes/MacHD/bsp.txt', '/Volumes/MacHD/lcbsp.txt').  
yes
```

gemacht werden.

Unter <http://www.cl.unizh.ch/sicemat/lehre/ws0405/pcl1/bsp.txt> findest du die Beispieldatei.