

3. Praktische Übung zur

## Logische und funktionale Programmierung

### GRUPPENÜBUNGEN:

#### (G 10)

Welcher ist der Unterschied zwischen den folgenden PROLOG - Programme?

```
add(0,X,X).
```

```
add(X,Y,Z):- X1 is X-1, add(X1,Y,Z1), Z is Z1+1.
```

und

```
add(0,Y,Y).
```

```
add(s(X),Y,s(Z)):-add(X,Y,Z).
```

Was lernen wir daraus?

#### (G 11)

Schreibe ein Prolog Programm, das alle Stationen der Klausenburger Linie 6 kennt.

Hinweis: Die Stationen sollen als Fakten unter dem Prädikat `station_6` gespeichert sein. Wie lautet die Frage an das Prologsystem um zu fragen ob `memorandumului` eine Station ist? Wie müssen Sie Prolog anfragen, um alle Stationen der Linie 6 auszugeben? In welcher Reihenfolge werden die Stationen ausgegeben?

#### (G 12)

Füge in dein Programm die Haltestellen der Linie 4 als Fakten ein. Definiere eine Regel `gemeinsamehaltestelle(X)`:- die erfüllt ist wenn sowohl die Linie 6 als auch die Linie 4 an der entsprechenden Haltestelle anhalten. Lasse Prolog alle gemeinsamen Haltestellen ausgeben.

#### (G 13)

Gegeben sei folgende Datenbasis mit den Prädikaten `maennlich/1`, `weiblich/1`, `vater/2`, `mutter/2`.

```
maennlich(paul). % Paul ist männlich
```

```
maennlich(fritz). % Fritz ist männlich
```

```
maennlich(steffen).% Steffen ist männlich
```

```
weiblich(karin). % Karin ist weiblich
```

```
weiblich(lisa). % Lisa ist weiblich
```

```
weiblich(maria). % Maria ist weiblich
```

```

weiblich(sina). % Sina ist weiblich
vater(steffen, paul). % Steffen ist der Vater von Paul
vater(fritz, karin). % Fritz ist der Vater von Karin
vater(steffen, lisa). % Steffen ist der Vater von Lisa
vater(paul, maria). % Paul ist der Vater von Maria
mutter(karin, maria). % Karin ist die Mutter von Maria
mutter(sina, paul). % Sina ist die Mutter von Paul

```

Definiere, darauf aufbauend, folgende Prädikate: `paar(M,F)`. Ein Mann  $M$  und eine Frau  $F$  sind ein Paar, wenn Sie beide Eltern eines gemeinsamen Kindes sind.

`geschwister(A,B)`.  $A$  und  $B$  sind Geschwister, wenn sie Kinder des gleichen Paares sind.

`neffe(N,X)`.  $N$  ist der Sohn der Schwester oder des Bruders von  $X$ .

#### (G 14)

Benutze die rekursive Definition der Addition, um zwei neue Prädikate zu schreiben `kleiner_oder_gleich` und `echt_kleiner`, die genau dann wahr sind, wenn  $X \leq Y$  und  $A < B$ .

#### (G 15)

Benutze die obigen Prädikate, um ein Prädikat `mod(X, Y, Z)` zu definieren, welches genau dann wahr ist, wenn  $X \bmod Y = Z$ .

#### (G 16)

Definiere Prädikate `und/2`, `oder/2`, `nund/2`, `noder/2`, `xor/2`, `impl/2` und `equ/2` (für die logische Äquivalenz) welche `wahr` oder `falsch` sind genau dann, wenn die entsprechenden logischen Operationen `wahr` oder `falsch` sind. Zum Beispiel, `und(A, B)` ist `wahr` genau dann, wenn sowohl  $A$  als auch  $B$  `wahr` sind. Achtung,  $A$  und  $B$  können selber Prolog goals sein (und nicht unbedingt Konstanten)!

Ein logischer Ausdruck in zwei Variablen wird in der Präfix Notation geschrieben, zum Beispiel so: `und(oder(A,B), nund(A,B))`.

Schreibe ein Prädikat `tafel/3` welches die Wahrheitstafel eines logischen Ausdruckes in zwei Variablen angibt.

BEISPIEL:

```
?- tafel(A, B, und(A, oder(A,B))).
```

```
wahr wahr wahr
```

```
wahr falsch wahr
```

```
falsch wahr falsch
```

```
falsch falsch falsch
```