

Übungen
Deskriptive Programmierung
SS 2008

Blatt 2

Aufgabe 1 (Prolog-Tutorial).

Gegeben seien folgende Prolog-Fakten der Form `person(Name,Alter,Vermögen)`,
`preis(Gegenstand,Preis)`, `mag(Person,Gegenstand)` und `besitzt(Person,Gegenstand)`:

```
person(james, 9, 2).      mag(james, fussball).  
person(john, 8, 8).      mag(john, fussball).  
person(harold, 10, 20).  mag(harold, buch).  
                        mag(harold, skateboard).  
  
preis(fussball, 5).      besitzt(james, fussball).  
preis(skateboard, 10).   besitzt(harold, fussball).  
preis(drachen, 3).      besitzt(harold, drachen).  
preis(buch, 6).         besitzt(john, skateboard).
```

a) Überlegen Sie sich Prolog-Anfragen zu den folgenden Fragen:

1. Besitzt John ein Buch?
2. Wie alt ist James?
3. Wer mag Fussbälle?
4. Wer besitzt einen Fussball?
5. Welche Gegenstände besitzt Harold?
6. Welche Gegenstände kann sich James leisten?
7. Besitzt Harold etwas, das James mag?
8. Besitzt Harold etwas, das mehr als 5 kostet?

b) Überlegen Sie sich Prolog-Regeln zu folgenden Prädikaten:

1. `aelter_als/2`, wobei `aelter_als(person1, person2)` genau dann gelten soll, wenn `person1` älter ist als `person2`.
2. `kann_sich_leisten/2`, wobei `kann_sich_leisten(person, gegenstand)` genau dann gelten soll, wenn sich die Person `person` den Gegenstand `gegenstand` leisten kann.
3. `moechte_haben/2`, wobei `moechte_haben(person, gegenstand)` genau dann gelten soll, wenn die Person `person` den Gegenstand `gegenstand` mag, aber nicht besitzt.
4. `gib/2`, wobei durch `gib(person, gegenstand)` das entsprechende `besitzt`-Fakt zu der Prolog-Faktenbasis hinzugefügt werden soll.
5. `kaufe/2`, wobei durch `kaufe(person, gegenstand)` zunächst überprüft werden soll, ob sich die Person `person` den Gegenstand `gegenstand` leisten kann, und gegebenenfalls anschließend die Prolog-Faktenbasis entsprechend modifiziert werden soll.
6. `tausche_um_zu_bekommen/2`, wobei durch `tausche_um_zu_bekommen(person, gegenstand)` ein Tausch zweier Gegenstände zwischen der Person `person` und einer anderen Person durchgeführt werden (d. h. die Prolog-Faktenbasis entsprechend abgeändert werden) soll, durch den `person` in den Besitz des Gegenstandes `gegenstand` gelangt. Diese andere Person muss `gegenstand` natürlich besitzen. Zudem wird niemand einen Gegenstand weggeben, den er mag und niemand darf durch einen Tausch einen Gegenstand bekommen, den er schon besitzt.
7. `erlange/2`, wobei durch `erlange(person, gegenstand)` die Person `person` versucht, in den Besitz des Gegenstandes `gegenstand` zu gelangen, indem sie zunächst probiert, ihn zu kaufen, und wenn das fehlschlägt, versucht, ihn durch einen Tausch zu erlangen.

Aufgabe 2 (Haskell-Tutorial*). Gegeben seien folgende Funktionsdefinitionen in Haskell:

```
a = 9
x = [1 .. a]
y = [a, a-1 .. 1]
```

Welche Bedeutung haben die folgenden Ausdrücke?

- `[1 | True]`, `[1 | False]`
- `[a | a > 0]`, `[a | a < 0]`
- `[z | z <- x, z <- y]`
- `[z | z <- x, b <- [1 .. z]]`, `[z | z <- x, c <- [1 .. a]]`
- `[z | z <- x, z `mod` 2 == 0]`, `[z | z <- x, a `mod` 2 == 0]`
- `[a | a `mod` 2 == 0, a <- x]`, `[a | a `mod` 2 == 1, a <- x]`
- `[[z | z <- x] | b <- y]`, `[z | z <- [b | b <- x]]`

Aufgabe 3 (Haskell-Tutorial). In der Vorlesung haben Sie gesehen, dass abzählbare (und insbesondere endliche) Mengen in Haskell durch Listen dargestellt werden können. Zudem wurden bereits einige Haskell-Implementierungen für verschiedene Operatoren der Mengenalgebra angegeben, wie z. B. Vereinigung, Durchschnitt, Differenz und Produkt. Implementieren Sie nun weitere Mengenoperatoren mittels folgender Funktionen:

- a) Eine zweistellige boolesche Funktion `subset m n`, die genau dann den Wert `True` liefert, wenn `m` eine Teilmenge von `n` ist.
- b) Eine zweistellige boolesche Funktion `equal m n`, die genau dann den Wert `True` liefert, wenn die Mengen `m` und `n` identisch sind.
- c) Eine einstellige Funktion `powset n`, die die Potenzmenge einer gegebenen Menge `n` berechnet.

Aufgabe 4 (Haskell-Tutorial). Definieren Sie eine rekursive Haskell-Funktion, die die Quersumme einer Zahl berechnet.