

Aufgabe 10 (2 Punkte) [Unifikation, Theorie]

- a) Überlegen Sie sich drei Terme t_1 , t_2 und t_3 , dabei soll t_1 mit t_2 , t_2 mit t_3 , aber t_3 nicht mit t_1 unifizierbar sein.
- b) Geben Sie für jeden Term und jeweils die anderen Terme an, ob sie miteinander unifizierbar sind und falls ja, wie der allgemeinste Unifikator aussieht.
 - 1) $f(x,y)$
 - 2) $f(z,z)$
 - 3) $f(g(f(x,y)),g(v))$
 - 4) $f(g(f(y,y)),g(x))$

Aufgabe 11 (1 Punkt)

Gegeben seien die folgenden, auch von der Veranstaltungsseite herunterladbaren Fakten. Bei den Anfragen ist es kein Problem, wenn einzelne Ausgaben mehrfach erscheinen.

```
%film(Film-Id, Filmname, FSK ab)
```

```
film(42, 'Wombat', 16).  
film(43, 'Schnuddel', 0).  
film(44, 'Hase', 18).  
film(45, 'Wut', 0).  
film(46, 'Wurmi', 16).  
film(47, 'Schnuddel', 18).
```

```
%kino(Name, Ort, Film-Id eines aktuell gezeigten Films)
```

```
kino(gloria, os, 42).  
kino(gloria, os, 43).  
kino(gloria, os, 47).  
kino(gloria, ol, 42).  
kino(gloria, ol, 44).  
kino(apollo, os, 43).  
kino(apollo, os, 45).
```

- a) Schreiben Sie eine Anfrage, die nacheinander (durch spätere Eingabe eines Semikolons) die Namen aller Filme mit FSK 16 ausgibt.
- b) Schreiben Sie eine Anfrage, die u. a. die Namen aller Filme mit einem FSK 16 oder höher ausgibt. (Das „unter anderem“ bezieht sich darauf, dass Ihre Ausgabe wahrscheinlich die Belegungen weiterer Variablen enthält.)
- c) Schreiben Sie eine Anfrage, die die Namen aller Kinos ausgibt, in denen ein Film mit dem Namen ‚Schnuddel‘ läuft.

Aufgabe 12 (2 Punkte)

Gegeben seien die in der Datei AufgabeStammbaum.pl gegebenen Fakten, die von Veranstaltungsseite geladen werden können. Die Fakten stehen nachfolgend auf der rechten Seite. Schreiben Sie folgende Anfragen bzw. das geforderte Prädikat

- a) Welche Personen sind nach 2000 geboren?

- b) Wie heißen die Kinder von Carl Gustav (generelle vereinfachende Annahme, er ist verheiratet mit der Frau, die Mutter des Kindes ist)?
- c) Schreiben Sie ein zweistelliges Prädikat, das true liefert, wenn die erste Person Oma oder Opa der zweiten Person ist? (Hinweis: ein Prädikat kann durch mehrere Regeln definiert werden, es sollte 16 Antworten eventuell in anderer Reihenfolge geben.)
?- oma_oder_opa(O,E).
- ```
O = 'Carl Gustav',
E = 'Alexander' ;
O = 'Carl Gustav',
E = 'Gabriel' ;
O = 'Silvia',
E = 'Alexander' ;
O = 'Silvia',
E = 'Gabriel' ;
O = 'Carl Gustav',
E = 'Oscar' ;
O = 'Carl Gustav',
E = 'Estelle' ;
O = 'Carl Gustav',
E = 'Leonore' ;
O = 'Carl Gustav',
E = 'Adrienne' ;
O = 'Carl Gustav',
E = 'Nicolas' ;
O = 'Daniel',
E = 'Dipsy' ;
O = 'Silvia',
E = 'Oscar' ;
O = 'Silvia',
E = 'Estelle' ;
O = 'Silvia',
E = 'Leonore' ;
O = 'Silvia',
E = 'Adrienne' ;
O = 'Silvia',
E = 'Nicolas' ;
O = 'Victoria',
E = 'Dipsy' ;
false.
```

```
person('Carl Gustav', 1946).
person('Silvia', 1943).
person('Daniel', 1973).
person('Christopher', 1974).
person('Victoria', 1977).
person('Carl Philip', 1979).
person('Sofia', 1984).
person('Madeleine', 1982).
person('Nicolas', 2015).
person('Adrienne', 2018).
person('Dipsy', 2035).
person('Estelle', 2012).
person('Oscar', 2016).
person('Alexander', 2016).
person('Gabriel', 2017).
person('Leonore', 2014).

verheiratet('Carl Gustav', 'Silvia').
verheiratet('Victoria', 'Daniel').
verheiratet('Carl Philip', 'Sofia').
verheiratet('Madeleine', 'Christopher').

mutterVon('Silvia', 'Victoria').
mutterVon('Silvia', 'Carl Philip').
mutterVon('Silvia', 'Madeleine').
mutterVon('Victoria', 'Oscar').
mutterVon('Victoria', 'Estelle').
mutterVon('Sofia', 'Alexander').
mutterVon('Sofia', 'Gabriel').
mutterVon('Madeleine', 'Leonore').
mutterVon('Madeleine', 'Adrienne').
mutterVon('Madeleine', 'Nicolas').
mutterVon('Estelle', 'Dipsy').
```