Übungsblatt 12 zu Programmiersprachenkonzepte

WS 2012/2013

Ausgabe:9.1.13

Abgabe bis zum Testattermin,
spätestens am 16.1.2013, 8 Uhr

Aufgabe 1 (2 Punkte):

Erstellen Sie Klauseln *mypraefix* mit 2 Listen als Argumenten so, dass die Klauseln beweisbar sind, wenn die erste Liste Präfix der zweiten Liste ist.

```
Bsp:        ?- mypraefix([a,b],[a,b,c,d]).
            true.
            ?- mypraefix([d],[a,b,c]).
            false.
            ?- mypraefix(X,[a,b]).
            X = [] ;
            X = [a] ;
            X = [a, b] ;
            false.
```

Aufgabe 2 (4 Punkte):

Erstellen Sie 3-stellige Klauseln *mycount* so, dass das erste Argument ein Wert ist, das zweite eine Liste und das Prädikat *mycount* beweisbar ist, wenn das dritte Argument die Anzahl des Auftretens des Wertes in der Liste ist.

```
Bsp:        ?- mycount(a, [a , b, c, a],2).
            true.
            ?- mycount(b, [a,b,c],X).
            X = 1 ;
            false.
```

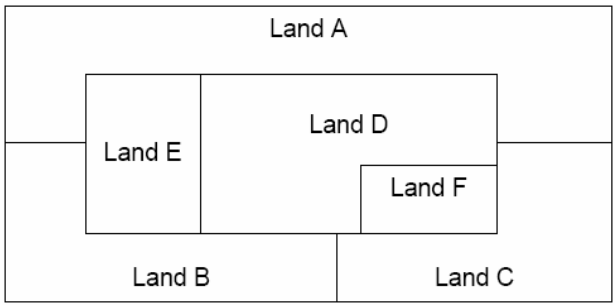
Aufgabe 3 (3 Punkte):

Erstellen Sie 3-stellige Klauseln *nachbar* so, dass die ersten beiden Argumente Werte sind, das dritte Argument eine Liste ist und die Klausel beweisbar ist, wenn die beiden Werte in der Liste auftreten und dort benachbart sind. Wenn benötigt, dürfen Sie sich weitere Hilfsklauseln erstellen.

```
Bsp:        ?- nachbar(a,b,[y,a,b,c,d]).
            true.
            ?- nachbar(a,c,[a,b,c]).
            false.
            ?- nachbar(b,X,[a,b,c]).
            X = c ;
            X = a ;
            false.
```

Aufgabe 4 (4 Punkte):

Gegeben Sei folgende Karte mit den 6 Ländern A,B,C,D,E, und F. Ihnen stehen vier Farben (rot, gelb grün, blau) zur Verfügung, um die Länder der Karte so einzufärben, dass aneinandergrenzende Länder unterschiedliche Farben haben.



Lösen Sie das Problem mit einem Prolog-Programm. Wie viele Färbungslösungen liefert Ihnen Ihr Prologprogramm?

Tipp: Ermitteln Sie die Lösung durch die Anfrage `?- karte(A,B,C,D,E,F).`
Versuchen Sie die Topologie der Karte durch `neben/2`–Klauseln zu beschreiben.