

Programmierung WS 18

Hausaufgaben - Blatt 10

Julian Giesen (MNR 388487)
Levin Gäher (MNR 395035)
Gruppe 12

HA 2

a)

```
istUnterkunft(wh).  
istUnterkunft(bh).  
istUnterkunft(sh).  
istUnterkunft(ph).  
istUnterkunft(jh).  
istUnterkunft(fb).  
  
hatEinenStern(ph).  
hatZweiSterne(sh).  
hatDreiSterne(jh).  
hatVierSterne(wh).  
hatFuenfSterne(fb).  
hatFuenfSterne(bh).
```

b)

```
?- hatFuenfSterne(X).
```

c)

```
hatEinenSternWeniger(X,Y) :- hatVierSterne(X), hatFuenfSterne(Y).  
hatEinenSternWeniger(X,Y) :- hatDreiSterne(X), hatVierSterne(Y).  
hatEinenSternWeniger(X,Y) :- hatZweiSterne(X), hatDreiSterne(Y).  
hatEinenSternWeniger(X,Y) :- hatEinenStern(X), hatZweiSterne(Y).
```

d)

```
?- hatEinenSternWeniger(_,X).
```

e)

```
hatWenigerSterne(X,Y) :- hatEinenSternWeniger(X,Y).  
hatWenigerSterne(X,Y) :- hatEinenSternWeniger(X,Z), hatWenigerSterne(Z,Y).
```

HA 4

a)

```
isList(nil).  
isList(cons(_, Rest)) :- isList(Rest).
```

b)

```
toPrologList(nil, []).  
toPrologList(cons(N, Rest), [Head|Tail]) :- Head = N, toPrologList(Rest, Tail).
```

c)

```
flatten([], []).  
flatten([[]|Res], YS) :- flatten(Res, YS).  
flatten([[X|XS]|Res], [X|YS]) :- flatten([XS|Res], YS).
```

d)

```
appendElement(X, Y, Z) :- flatten([X, [Y]], Z).
```

e)

```
reverseList([], []).  
reverseList([X], [X]).  
reverseList([X|XS], Y) :- reverseList(XS, Z), appendElement(Z, X, Y).
```
