Logic week3

Michelle Bergin

November 3, 2018

1 Practical session in class

If this is submitted as a zip, any .pl's are added in the zip

2 4.1-4.7

4.1 How does Prolog respond to the following queries?

1 [a,b,c,d] = [a,[b,c,d]].

no.

2 [a,b,c,d] = [a|[b,c,d]].

yes.

3 [a,b,c,d] = [a,b,[c,d]].

no.

4 [a,b,c,d] = [a,b|[c,d]].

yes.

5 [a,b,c,d] = [a,b,c,[d]].

no.

6 [a,b,c,d] = [a,b,c|[d]].

yes.

7 [a,b,c,d] = [a,b,c,d,[]].

no.

8 [a,b,c,d] = [a,b,c,d|[ ]].

yes.

9 [ ] = _.

yes.

10 [ ] = [_].

no.

11 [ ] = [_|[ ]].

no.

4.2 Which of the following are syntactically correct lists? If the representation is correct, how many elements does the list have?

1 [1|[2,3,4]]

L = 4

2 [1,2,3|[ ]]

L = 3

3 [1|2,3,4]

Incorrect Syntax

4 [1|[2|[3|[4]]]]

L = 4

5 [1,2,3,4|[ ]]

L = 4

6 [[ ]|[ ]]

L = 1

7 [[1,2]|4]

Incorrect Syntax

8 [[1,2],[3,4]|[5,6,7]]

L = 5

4.3 Write a predicate second(X,List) which checks whether X is the second element of List .

```
second(X,[_,[X|_]).
```

4.4 Write a predicate swap12(List1,List2) which checks whether List1 is identical to List2 , except that the first two elements are exchanged.

```
swap12([A,B|T],[B,A|T]).
```

1. Suppose we are given a knowledge base with the following facts:

```
tran(eins,one).
tran(zwei,two).
tran(drei,three).
tran(vier,four).
tran(fuenf,five).
tran(sechs,six).
tran(sieben,seven).
tran(acht,eight).
tran(neun,nine).
```

Write a predicate listtran(G,E) which translates a list of German number words to the corresponding list of English number words. For example:

```
listtran([eins,neun,zwei],X).
```

should give:

```
X = [one,nine,two].
```

Your program should also work in the other direction. For example, if you give it the query

```
listtran(X,[one,seven,six,two]).
```

it should return:

```
X = [eins,sieben,sechs,zwei].
```

(Hint: to answer this question, first ask yourself How do I translate the empty list of number words?. Thats the base case. For non-empty lists, first translate the head of the list, then use recursion to translate the tail.)

4.7 Draw the search trees for the following three queries:

```
?- member(a,[c,b,a,y]).
```

member(a,[c|_]).
false.
member(a,[_|[b,a,y]]):- member(a,[b,a,y]).
member(a,[b_]).
false.
member(a,[_|[a,y]]):- member(a,[a,y]).

member(a,[a|_]).

true.

```
?- member(x,[a,b,c]).
```

---

member(x,[a|_]).

false.

member(x,[_|[b,c]]):- member(x,[b,c]).

member(x,[b|_]).

false.

member(x,[_|[c]]):- member(x,[c]).

member(x,[c|[ ]]).

false.

```
?- member(X,[a,b,c]).
```

---

member(a,[a|_]).

X = a.

member(X,[_|[b,c]]):- member(X,[b,c]).

member(b,[b|_]).

X = b.

member(X,[_|[c]]):- member(X,[c]).

member(c,[c|_]).

X = c.