02156 Exercises-03

Jørgen Villadsen

2021

Exercise 1

Express the predicate `member` using the predicate `append` only (no recursion allowed). It is pretty simple. Call the defined predicate `membera` (`member` via `append`). Test it carefully.

Exercise 2

First load the file `logic.pl` with the program `truthtable` from CampusNet (Program files folder in the top folder).

The program implements truth tables for propositional logic using the following operators:

```
:- op(650,xfy,\<=>). /* difference  */
:- op(650,xfy,<=>).  /* equivalence */
:- op(640,xfy,=>).   /* implication */
:- op(630,xfy,\).    /* disjunction */
:- op(620,xfy,&).    /* conjunction */
:- op(610,fy, ~).    /* negation    */
```

Note that xor is called difference here.

Example: $(p \oplus q) \leftrightarrow (\neg(p \rightarrow q) \vee \neg(q \rightarrow p))$

```
?- truthtable( (p \<=> q) <=> (~ (p => q) \ ~ (q => p)) ).
(p\<=>q)<=> ~ (p=>q)\ ~ (q=>p)  p q  value
                                t t    t
                                t f    t
                                f t    t
                                f f    t

Yes
```

Use the program to determine which of the following formulas are valid and which are not valid.

$$p \rightarrow q \rightarrow p$$

$$(p \rightarrow q) \rightarrow p$$

$$(p \rightarrow q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$$

$$p \rightarrow (q \rightarrow r) \rightarrow (p \rightarrow q) \rightarrow p \rightarrow r$$

$$((p \rightarrow q) \rightarrow p) \rightarrow p$$

$$p \rightarrow q \rightarrow p \rightarrow p$$

Draw the trees for the formulas if necessary in order to understand the syntax.

Exercise 3

Show that $(p \oplus q) \oplus q \equiv p$ using the following logical equivalences only:

$$A \oplus \mathit{false} \equiv A$$

$$A \oplus A \equiv \mathit{false}$$

$$A \oplus (B \oplus C) \equiv (A \oplus B) \oplus C$$

Exercise 4

Construct a completed tableau that shows whether the formula $p \to q \to p$ is valid.

Does the tableau show that the formula is valid?

Exercise 5

Construct a completed tableau that shows whether the formula $(p \to q) \to p$ is valid.

Does the tableau show that the formula is valid?

Exercise 6

Write a program `qs(+List1,?List2)` that succeeds if and only if `List2` is a permutation of `List1` and the elements in `List2` are ordered by the `=<` relation.

Use the Quicksort algorithm and partition on the first element in a non-empty list.

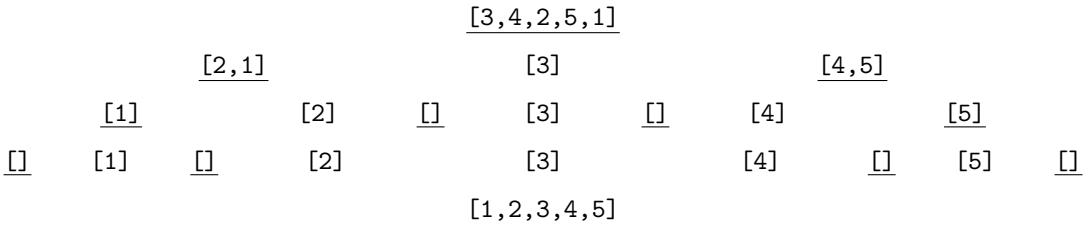
Quicksort sorts the list by making two recursive calls and combining the results (unless the list is empty): one call with the elements smaller than the partition element and one call with the elements larger than the partition element.

Efficiency is not a concern here and predicates like `member` and `append` can be used.

Use the following program `part` to do the partitioning.

```
part(_,[],[],[]).
part(X,[Y|Xs],[Y|Ls],Bs) :- X > Y, part(X,Xs,Ls,Bs).
part(X,[Y|Xs],Ls,[Y|Bs]) :- X =< Y, part(X,Xs,Ls,Bs).
```

Consider the following example (calls to `qs` are underlined):



More information about Quicksort is available on Wikipedia. :-)

Finally please study the predicates in the file `logic.pl` in details but ignore the special `!` predicate which will be explained next week.

In particular try to understand the following query:

```
?- generate([p,q],V).

V = [ (p, t), (q, t)] ;

V = [ (p, t), (q, f)] ;

V = [ (p, f), (q, t)] ;

V = [ (p, f), (q, f)] ;

No
```

It will be used next week.