Meta–Interpreter 1:

```
prove(true) :- !.
prove((Goal1, Goal2)) :- !, prove(Goal1), prove(Goal2).
prove(Goal) :- clause(Goal, Body), prove(Body).
```

Meta–Interpreter 2:

```
prove(true) :- !.
prove((Goal1, Goal2)) :- !, prove(Goal2), prove(Goal1).
prove(Goal) :- clause(Goal, Body), prove(Body).
```

Meta–Interpreter 3:

```
prove(true,0) :- !.
prove((Goal1,Goal2),N) :- !, prove(Goal1,N1), prove(Goal2,N2),
                          N is N1+N2.
prove(Goal,N) :- clause(Goal, Body), prove(Body,N1),
                 N is N1+1.
```

Klassisches `append`:

```
append([],Ys,Ys).
append([X|Xs],Ys,[X|Zs]) :- append(Xs,Ys,Zs).
```

`app` mit Differenzlisten:

```
app(Xs - Ys, Ys - Zs, Xs - Zs).
```

Beispielgrammatik $G = (N, T, S, P)$

- $N =$ { Satz, Nominalphrase, Verbalphrase, Artikel, Nomen, Verb }
- $T =$ { a, the, cat, mouse, scares, hates }
- $S =$ Satz
- P besteht aus folgenden Regeln:

Satz	$\longrightarrow$	Nominalphrase Verbalphrase
Nominalphrase	$\longrightarrow$	Artikel Nomen
Verbalphrase	$\longrightarrow$	Verb
Verbalphrase	$\longrightarrow$	Verb Nominalphrase
Artikel	$\longrightarrow$	a
Artikel	$\longrightarrow$	the
Nomen	$\longrightarrow$	cat
Nomen	$\longrightarrow$	mouse
Verb	$\longrightarrow$	scares
Verb	$\longrightarrow$	hates

$L(G)$ enthält: `a cat scares the mouse`, `the mouse hates the cat`,
`a mouse scares a mouse`, `a mouse hates`

Beispielgrammatik in Prolog

```
satz --> nominalphrase, verbalphrase.
nominalphrase --> artikel, nomen.
verbalphrase --> verb.
verbalphrase --> verb, nominalphrase.
artikel --> [a].
artikel --> [the].
nomen --> [cat].
nomen --> [mouse].
verb --> [scares].
verb --> [hates].
```

Überführung in Prolog–Klauseln

```
satz(S, R) :- nominalphrase(S, VP), verbalphrase(VP, R)
nominalphrase(NP, R) :- artikel(NP, N), nomen(N, R).
verbalphrase(VP, R) :- verb(VP, R).
verbalphrase(VP, R) :- verb(VP, NP), nominalphrase(NP, R).
artikel([a | R], R).
artikel([the | R], R).
nomen([cat | R], R).
nomen([mouse | R], R).
verb([scares | R], R).
verb([hates | R], R).
```