

Prolog: Einsatz vor allem  
in der KI (insbes.  
für Expertensysteme,  
Robotik, ...)

Prinzip des logischen  
Programmierens:

- Programm = Wissensbasis,  
beschreibt logische  
Zusammenhänge des  
zu lösenden Problems
- Ausführung d. Prog:  
Anfragen an das Prog.  
werden beantwortet/  
bewiesen.
- Prolog-Programm besteht  
aus bestimmten logischen

Formeln (sogenannte  
Klauseln):

- Fakten
- Regeln

• Faktum:

- Name der Eigenschaft  
bzw. Relation
- in runden Klammern  
folgen die Objekte,  
die die Eigenschaft  
haben
- zum Schluss: .

• Relationen können versch.  
Stelligkeit haben und  
sie sind gerichtet.

• Relationsname = Prädik-  
katssymbole

• Prädikatssymbole u.,

Objekte müssen mit  
Kleinbuchstaben beginnen

- Kommentare in  
Prolog:

% ... Zeilenende

oder

/\*

⋮

\*/

- Anfragen beginnen  
mit:

?- ... .

- Prolog verwendet die  
Closed World

Assumption:

Alles, was nicht aus  
der Wissensbasis

folgt, ist falsch.

?- verheiratet(monika,  
werner).

false

## Variablen im Programm

- Variablen im Prolog beginnen mit Großbuchstaben oder mit \_.
- Variablen im Programm sind allquantifiziert (d.h. sie stehen für beliebige Belegungen).

?- menschl(5).

true

- Mehrfache Vorkommen

der gleichen Variable  
in der gleichen Klausel  
müssen auch gleich  
belegt werden.

Bsp:

Ergänze Prog. um  
folgendes Faktum:

$\text{mag}(X, X).$

(Jeder mag nur  
sich selbst.)

$\text{mag}(X, Y).$

(Jeder mag jeden.)

Gleiche Variablen in  
verschiedenen Klauseln  
haben nichts miteinander

Zu tun.

## Prolog-Implementierung

z.B. SWI-Prolog

### Variablen in Anfragen

- Prolog ist aus dem automatischen Beweisen entstanden und versucht, Anfragen zu beweisen.
- Prolog = Programming in Logic
- Prog. kann auch selbst Lösungen berechnen, wenn die Anfrage Variablen enthält.

- Variablen in Anfragen sind existenzquantifiziert.

Bsp: Gibt es ein X, das die Mutter von susanne ist?

bzw: Wer ist die Mutter von susanne?

Prolog liefert als Ergebnis eine Antwortsubstitution.

Sie gibt an, wie die Variablen belegt werden müssen, damit die Anfrage wahr wird.

? Bsp: Wer sind die Kinder von

renate?

- Ein- u. Ausgabe von Prädikaten liegt nicht fest, sondern hängt von der Anfrage ab. Ausgabe sind die Positionen, an denen Variablen in der Anfrage stehen.

z.B.: ?-mutter(X,Y).

- Prolog durchsucht die Wissensbasis von oben nach unten u. liefert die erste Antwort, die es findet.

Danach kann Benutzer

; ← steht für "oder"

eingeben, um Prolog nach weiteren Lösungen suchen zu lassen.

## Kombination von Anfragen

Bsp: Ist gerd der Vater von susanne?

F muss in der Klausel gleich belegt werden.

Prolog löst Anfragen, indem es Beweisbaum

Konstruiert: steht für "und"

?-verh(gerd, F), mV(F, sus)

F=renate

Prolog bearbeitet Anfragen von links nach rechts

?-mV(renate, susanne)

|  
□ ← leere Klausel,  
d.h. es muss  
nichts mehr  
bewiesen werden

Aus erfolgreichen Beweispfaden (von der ursprünglichen Anfrage bis zu □) wird dann die Antwortsubstitution abgelesen.

?-mV(O, M), mV(M, aline).

O=monika / O=monika / O=renate  
M=Karin / M=klaus / M=susanne

?-mV(Karin,  
aline).

?-mV(klaus,  
aline).

?-mV(  
susanne,  
aline)

↓

↓

|



Wenn der Beweisversuch scheitert, dann setzt Prolog zum Knoten darüber im Beweisbaum zurück und versucht, weitere Lösungen zu finden (Backtracking).

Beweisbaum wird in Tiefensuche von links nach rechts aufgebaut. Suche stoppt, sobald Lösung gefunden wurde (u. kann mit ; fortgesetzt werden).

Arbeitsweise u. Effizienz von Prolog-Programmen

lassen sich beeinflussen,  
denn:

- Prop-Klauseln werden von oben nach unten abgearbeitet.
- Zusammengesetzte Klauseln werden von links nach rechts abgearbeitet.

## Regeln

- Neben Fakten kann das Prop. Regeln enthalten, um aus bekanntem Wissen neues Wissen herzuleiten.

$\therefore P :- q, r$

← bedeutet "falls"

Kopf  $\rightarrow$   $P := \overbrace{q, r}^{\text{Rumpf}}$ .

bedeutet:  $p$  gilt, falls  
 $q$  und  $r$  gelten.

bzw: Wenn  $q$  und  $r$   
gelten, dann gilt  $p$ .

?-  $vV(\text{gerd}, \text{susanne})$ .

|  $V = \text{gerd}$   
|  $K = \text{susanne}$

?-  $\text{verb}(\text{gerd}, F), mV(F, \text{susanne})$

|  $F = \text{renate}$

□

Antwortsubst. ist nur der  
Teil der Variablenbelegung,  
der die Variablen in der  
ursprünglichen Anfrage  
betrifft.

## Mehrere Regeln für das gleiche Prädikat

Damit lässt sich aus-  
drücken, dass  $p$  gilt,  
falls  $q$  oder  $r$  gilt:

$$P :- q.$$

$$P :- r.$$

?-elternteil(X, susanne).

/ y=susanne

\ y=susanne

?-mV(X, susanne)

?-vV(X, sus).

| X=renate

| X=gerd

□

□

## Rekursive Regeln

Prädikat "vorfahre":

- $V$  ist vorfahre von  $X$ , wenn  $V$  ein elternteil von  $X$  ist.
- $V$  ist vorfahre von  $X$ , wenn es ein  $Y$  gibt, so dass  $V$  elternteil von  $Y$  ist und  $Y$  vorfahre von  $X$  ist.

( $\Rightarrow$  führt zu rekursiver Regel)

