



## Outline

- 1 Einleitung
- 2 Einführung in C
- 3 Fortgeschrittenes in C
- 4 Einführung in Emacs Lisp
- 5 Einführung in Prolog



- ... vom Französischen: *Programmation en Logique*
- Anfang 1970er maßgeblich von Alain Colmerauer entwickelt
- erster ISO-Standard für Syntax erst 1995
- wichtigste logische Programmiersprache
- Beispiele für Anwendungsgebiete:
  - in 1980ern wichtige Rolle bei Entwicklung von Expertensystemen
  - in der Computerlinguistik
  - im Forschungsgebiet der künstlichen Intelligenz



- Wir verwenden den SWI-Prolog-Interpreter:  
<http://www.swi-prolog.org>
  - zum starten `swipl` (in den Computerpools mit `p1`)
  - - Hilfe: `help`.
    - Datei laden: `consult('pfad/dateiname')`.
      - Beachte die Benutzung von einfachen Hochkomma (')
    - Beenden: `halt`.
  - nach dem Laden einer Datei: wie gewohnt Anfragen stellbar
- ⇒ Ausgaben weiterer gültiger Belagungen mit "Leertaste"



## Prolog - Fakten, Regeln und Anfragen

- Sprachprimitive: Fakten und Regeln
- Wird *Wissensbasis* genannt
- Muss aus Datei geladen werden (also keine direkte Eingabe in Prolog-Interpreter über Kommandozeile möglich).
- Auf Wissensbasis können Anfragen gestellt werden.



## Prolog - Fakten und Regeln

- Wissensbasen bestehen aus Fakten und Regeln (an “.” denken, “:-” entspricht umgekehrten Implikationspfeil).

```
1 liebt(vincent, mia).  
2 liebt(marcellus, mia).  
3 liebt(pumpkin, honeybunny).  
4 liebt(honeybunny, pumpkin).  
5 eifersuechtig(X, Y) :- liebt(X, Z), liebt(Y, Z).
```

- Für Anfragen findet Prolog eine Variablenbelegung (Bedingungen), welche eine Anfrage erfüllen.

```
1 ?- liebt(vincent, mia).  
2 true.  
3 ?- liebt(X, mia).  
4 X=vincent ;  
5 X=marcellus ;  
6 false.
```



## Prolog - Anfragen

- Anfragen beginnen mit `?-` und enden auf einem *Punkt*.
- “Nicht-Operator” `\+` fordert die Nicht-Ableitbarkeit eines Terms

```
1 ist_mann(vincent).
2 ist_mann(marcellus).
3 ist_frau(mia).
4 ?- ist_frau(mia).
5 true.
6 ?- ist_mann(sokrates).
7 false.
8 ?- \+ist_mann(sokrates).
9 true.
```

- **Wichtig:** Prolog kann keine Aussagen über *wahr* und *falsch* machen, da es selbst kein Wissen darüber hat! Prolog kann nur anhand der gegebenen Fakten Aussagen darüber machen, ob etwas herleitbar/beweisbar ist oder nicht!



## Prolog - Anfragen

- Der Interpreter sucht systematisch in einer gegebenen Datenbasis nach Antwort auf eine Anfrage.
- Anfragen ohne Variablen:
  - Ein positives Resultat bedeutet, dass Anfrage logisch ableitbar war.
- Anfragen mit Variablen:
  - Als Antwort werden hier zusätzlich Belegungen angegeben, mit denen die Anfrage wahr wird.
- Prolog nutzt zum lösen von Anfragen eine Tiefensuche mit Backtracking.

- Beispiel:

1  $f(a)$ .

2  $f(b)$ .

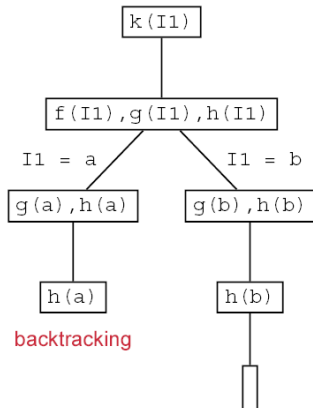
3  $g(a)$ .

4  $g(b)$ .

5  $h(b)$ .

6  $k(X) :- f(X), g(X), h(X)$ .

7  $?- k(X)$ .

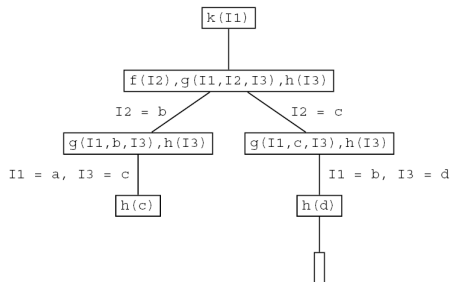






# Beweisen in Prolog

- 1 `f(b).`
- 2 `f(c).`
- 3 `g(a,b,c).`
- 4 `g(b,c,d).`
- 5 `h(d).`
- 6 `k(X):-f(Y),g(X,Y,Z),h(Z).`
- 7 `?- k(X).`





- Beispiele für die Grundlegenden Rechenoperationen.
  - X is  $8 + 3 \rightarrow x = 8 + 3$
  - X is  $2 - 1 \rightarrow x = 2 - 1$
  - X is  $3 * 5 \rightarrow x = 3 \cdot 5$
  - X is  $6 / 4 \rightarrow x = \frac{6}{4}$



- `is` ist ein spezielles Prädikat zum Auswerten von arithmetischen Ausdrücken. Zum Vergleich:

1 ?- `X = 8 + 3`

2 `X = 8 + 3.`

3 ?- `X is 8 + 3.`

4 `X = 11.`

- Achtung: Alle Variablen die rechts von `is` stehen müssen bereits unifiziert sein. Links von `is` kann eine Variable stehen, welche wenn nötig unifiziert wird.

1 ?- `8 + 3 is X.`

2 `ERROR: ...`

3 ?- `Y = 3, X is 8 + Y.`

4 `X = 11.`



- Beispiele für die Grundlegenden Vergleichsoperatoren:
  - $X < Y \rightarrow x < y$
  - $X > Y \rightarrow x > y$
  - $X \leq Y \rightarrow x \leq y$
  - $X \geq Y \rightarrow x \geq y$
  - $X ::= Y \rightarrow x = y$
  - $X \neq Y \rightarrow x \neq y$

Dabei sind  $X$  und  $Y$  Terme, deren Werte verglichen werden. Dafür müssen alle Variablen bereits unifiziert sein.



- Test auf Termäquivalenz:

- 1 ?-  $=(a, a)$ .
- 2 **true**.
- 3 ?-  $=(a, b)$ .
- 4 **false**.
- 5 ?-  $=(a, X)$ .
- 6  $X=a$ .
- 7 ?-  $X=Y, Y=a, X=b$ .
- 8 **false**.
- 9 ?-  $f(X, X)=f(a, b)$ .
- 10 **false**.
- 11 ?-  $f(r(a), X)=f(Y, z(b))$ .
- 12  $X=z(b)$ ,
- 13  $Y=r(a)$ .



- Im Folgenden schauen wir uns weitere Techniken zum Erzeugen beliebig vieler Regeln mit Hilfe der Rekursion und der dynamischen Vergrößerung der Datenstruktur durch Listen an.



Wir wollen Zahlen als Terme von  $0$  und  $\text{succ}$  beschreiben ...

- $0 \rightarrow 0$
- $1 \rightarrow \text{succ}(0)$
- $2 \rightarrow \text{succ}(\text{succ}(0))$
- $3 \rightarrow \text{succ}(\text{succ}(\text{succ}(0)))$
- ...

Das Prädikat `number` soll für all diese Ausdrücke “wahr” sein.

$1 \text{ number}(0) .$

$2 \text{ number}(\text{succ}(N)) :- \text{number}(N) .$

Wichtig: Abbruchbedingung! Sie sollte vor der Rekursion definiert werden.



- `[]` ist eine leere Liste.
- `[H|T]` ist eine Liste, wobei H für einen Term und T für eine Liste steht.
- Beispiel: Bestimmung der Länge einer Liste:

```
1     laenge([],0).  
2     laenge([_|T],N) :- laenge(T,M), N is M+1.
```

- *Bemerkung:* `_` ist eine spezielle Variable, deren Wert uninteressant ist und daher nicht ausgegeben wird.





## Rätselraten

$$\begin{array}{r} \text{ABB} - \text{CD} = \text{EED} \\ - \quad - \quad * \\ \text{FD} + \text{EF} = \text{CE} \\ = \quad = \quad = \\ \text{EGD} * \text{FH} = ??? \end{array}$$

A bis H stehen jeweils für eine Ziffer 0 bis 9, wobei nicht klar ist, welche Zahl an welchen Buchstaben gebunden ist. Gesucht ist die Zahl, die bei den Fragezeichen stehen muss.



## Einsteinsches Rätsel

- ... stammt angeblich von Einstein
  - angeblich nur 2% der Weltbevölkerung können es lösen
1. Es gibt fünf Häuser mit je einer anderen Farbe.
  2. In jedem Haus wohnt eine Person anderer Nationalität.
  3. Jeder Hausbewohner bevorzugt ein bestimmtes Getränk, raucht eine bestimmte Zigarettenmarke und hält ein bestimmtes Haustier.
  4. Keine der fünf Personen trinkt das gleiche Getränk, raucht die gleichen Zigaretten oder hält das gleiche Tier wie seine Nachbarn.
- ⇒ *Frage: Wem gehört der Fisch?*



## Einsteinsches Rätsel - Wissensbasis

- Der Brite lebt im roten Haus.
- Der Schwede hält einen Hund.
- Der Däne trinkt gern Tee.
- Das grüne Haus steht direkt links neben dem weißen Haus.
- Der Besitzer des grünen Hauses trinkt Kaffee.
- Die Person, die Pall Mall raucht, hält einen Vogel.
- Der Mann, der im mittleren Haus wohnt, trinkt Milch.
- Der Besitzer des gelben Hauses raucht Dunhill.
- Der Norweger wohnt im ersten Haus.
- Der Marlboro-Raucher wohnt neben dem, der eine Katze hält.
- Der Mann, der ein Pferd hält, wohnt neben dem, der Dunhill raucht.
- Der Winfield-Raucher trinkt gern Bier.
- Der Norweger wohnt neben dem blauen Haus.
- Der Deutsche raucht Rothmans.
- Der Marlboro-Raucher hat einen Nachbarn, der Wasser trinkt.