

## Zur Erinnerung: $C_0 \rightarrow H_0$

- ▶ Gegeben: Ein  $(m, k, i)$ - $C_0$ -Programm mit  $m$  Variablen,  $k$  Eingabevariablen und  $x_i$  als Ausgabe.
- ▶ Vor jedes Statement wird ein Ablaufpunkt gesetzt und diesem eine  $m$ -stellige Funktion  $f$  zugeordnet, die als Argumente genau die  $m$  Programmvariablen  $x_1$  bis  $x_m$  hat.
- ▶ Der Funktionswert wird mit Hilfe derjenigen Funktionen beschrieben, die den als nächstes erreichbaren Ablaufpunkten zugeordnet sind.

## Zur Erinnerung: $H_0 \rightarrow C_0$

- ▶ Grundidee: Umwandlung in ein `while`-Programm.
- ▶ Die Schleife wird solange durchlaufen, wie noch rekursive Funktionsaufrufe auszuführen sind (also `flag == 1` gilt).
- ▶ In der Schleife befinden sich die Übersetzungen der Funktionsdefinitionen. Die Auswahl der bei der nächsten Iteration auszuführenden Funktion erfolgt durch die Variable `function`, deren Wert in `if`-Anweisungen abgefragt wird.
- ▶ Das Ergebnis wird in der Variable `result` gespeichert und nach der `while`-Schleife mit `printf` ausgegeben.
- ▶ Da Parameterwerte voneinander abhängen können, kann es notwendig sein, dass Zuweisungen umsortiert oder gar Hilfsvariablen eingeführt werden müssen.

# Einführung in Prolog

- ▶ Prolog (vom Französischen: **Pro**grammation en **Log**ique, deutsch: **Pro**grammieren in **Log**ik)
- ▶ Prolog-Programme bestehen aus einer Datenbasis, deren Einträge **Fakten** und **Regeln** genannt werden.
- ▶ Der Nutzer formuliert **Anfragen**, auf die der Interpreter systematisch durch Nutzung dieser Fakten und Regeln eine Antwort findet.
- ▶ Ein positives Ergebnis bedeutet, dass die Anfrage ableitbar war.

# Einführung in Prolog: Fakten

- ▶ Albert ist männlich:  
`male(albert).`
- ▶ Berti ist ein Elternteil von Albert:  
`parent(berti,albert).`
- ▶ Ein Fakt besteht aus einem Prädikat und dessen Argumenten.
- ▶ Statements werden immer mit Punkt abgeschlossen.
- ▶ Variablen beginnen mit Großbuchstaben.
- ▶ Albert ist Kind aller:  
`parent(X,albert).`

# Einführung in Prolog: Regeln

- ▶ Alles, was nicht männlich ist, ist weiblich:  
`female(X) :- not(male(X)).`
- ▶ Eine Regel beschreibt die Abhängigkeit eines Fakts von einem oder mehreren anderen Fakten.
- ▶ Der Regeloperator `:-` ist dabei wie ein umgedrehter Implikationspfeil zu lesen.
- ▶ Das Komma kann dabei als Und-Operator (und das Semikolon als Oder-Operator) verwendet werden.
- ▶ Haben zwei Regeln die gleiche Konsequenz, folgt diese, wenn mindestens in einer Regel die Bedingung erfüllt ist.
- ▶ Wenn X Elternteil von Y und männlich ist, dann ist X Vater von Y: `father(X,Y) :- parent(X,Y), male(X).`

# Einführung in Prolog: Anfragen

- ▶ Ist Albert männlich?  
`?- male(albert).`
- ▶ Eltern von Albert?  
`?- parent(X,albert).`
- ▶ Alle Eltern mit Kindern?  
`?- parent(X,Y).`
- ▶ Anfragen werden im Prolog-Interpreter gestellt.
- ▶ In Anfragen kann ebenso mit Variablen gearbeitet werden, um erfüllenden Belegungen zu finden.