

depth ^{Ziele} (Goals, ^{Open} [X | -], ^{Closed} -) :-

member (X, Goals), !,
write (X).

----- Exkurs -----

? depth ([2], [a], []).

-> Anfrage, ob ich von a nach 2 komme

depth (Goals, [X | RestOpen], Closed) :-

member (X, Closed), !,
depth (Goals, RestOpen, Closed).

für Fall, dass wir X schon angeschaut haben

depth (Goals, [X | RestOpen], Closed) :-

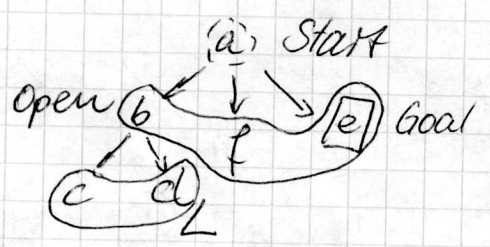
write (X),
Kinder (X, L), *Liste der Knotenkinder von X*

Fortsetzung

append (L, RestOpen, Open),

% L und RestOpen zusammen (= Open) müssen
% noch durchsucht werden

Bsp:



↓ Tiefensuche

Breitensuche
→

Breitensuche kann allein durch
append (RestOpen, L, Open)
definiert werden

depth depth (Goals, Open, [X | Closed]).

! append ist Listenkonnkatenation, ist in Prolog schon drin

? kinder (a, L).
L = [b, c].

