

Prolog Homework 1: search trees

answer

Backtracking

- Query: $k(Y)$

Homework:

1. write out the full set of search trees for this query to find out what Prolog should produce as solutions for Y .
2. check that Prolog produces the results you thought
3. see if you can follow through using `trace(k)` the steps that Prolog actually went through: are they the same as your proof tree?

Example

```
f(a).  
f(b).  
g(a).  
g(b).  
h(b).  
k(X):- f(X), g(X), h(X).
```

```
?- k(Y).
```

Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

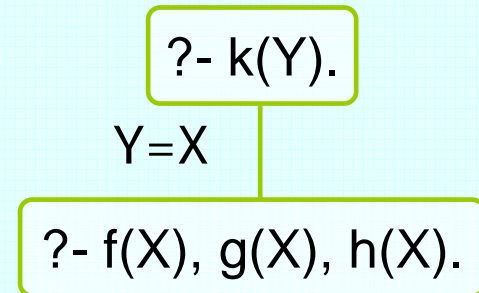
?- k(Y).

?- k(Y).

Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

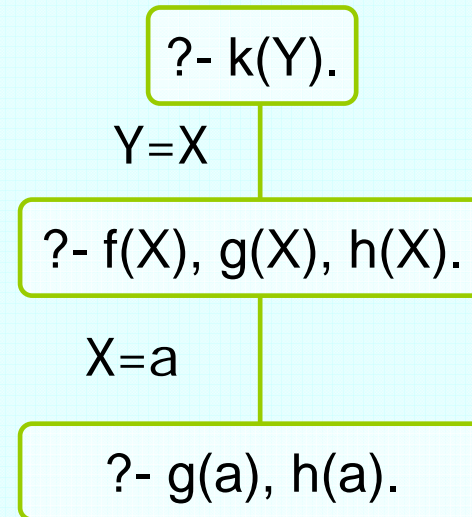
?- k(Y).



Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

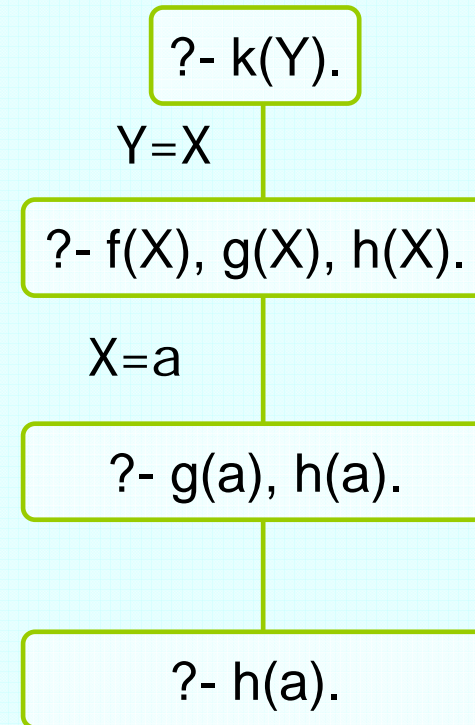
?- k(Y).



Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

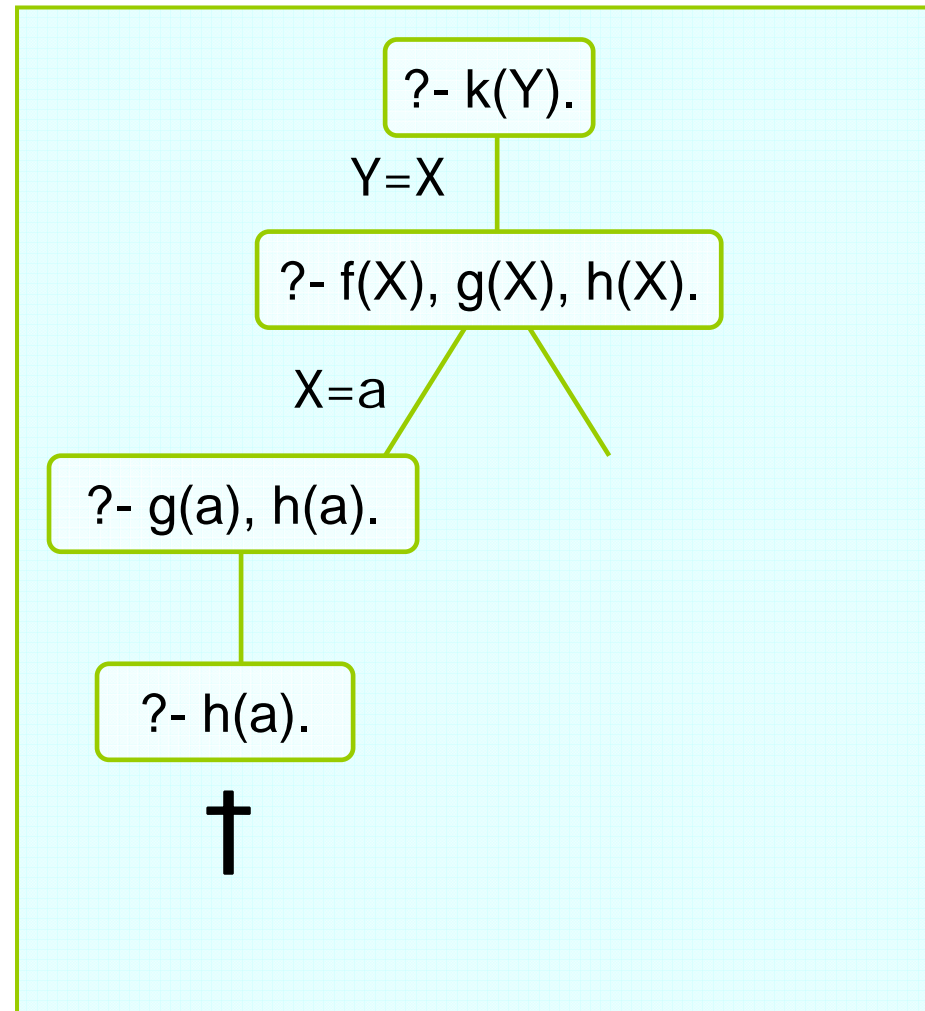
?- k(Y).



Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

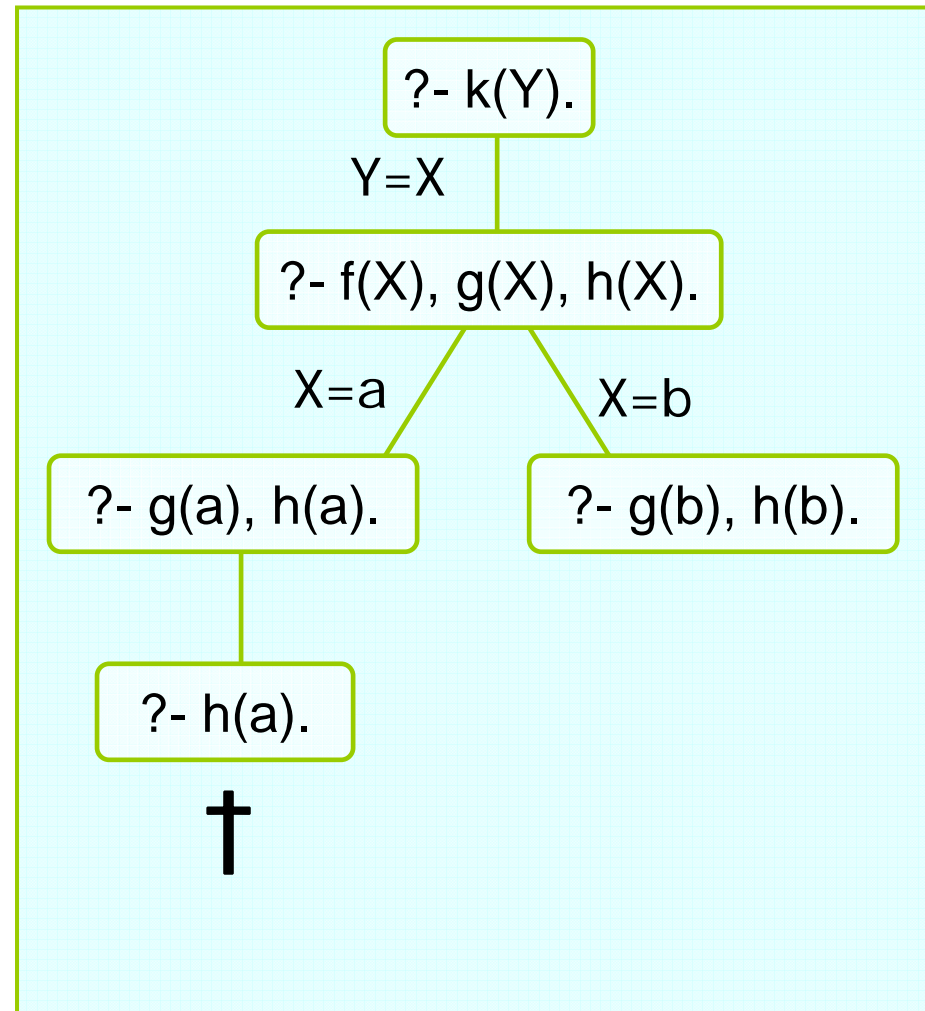
?- k(Y).



Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

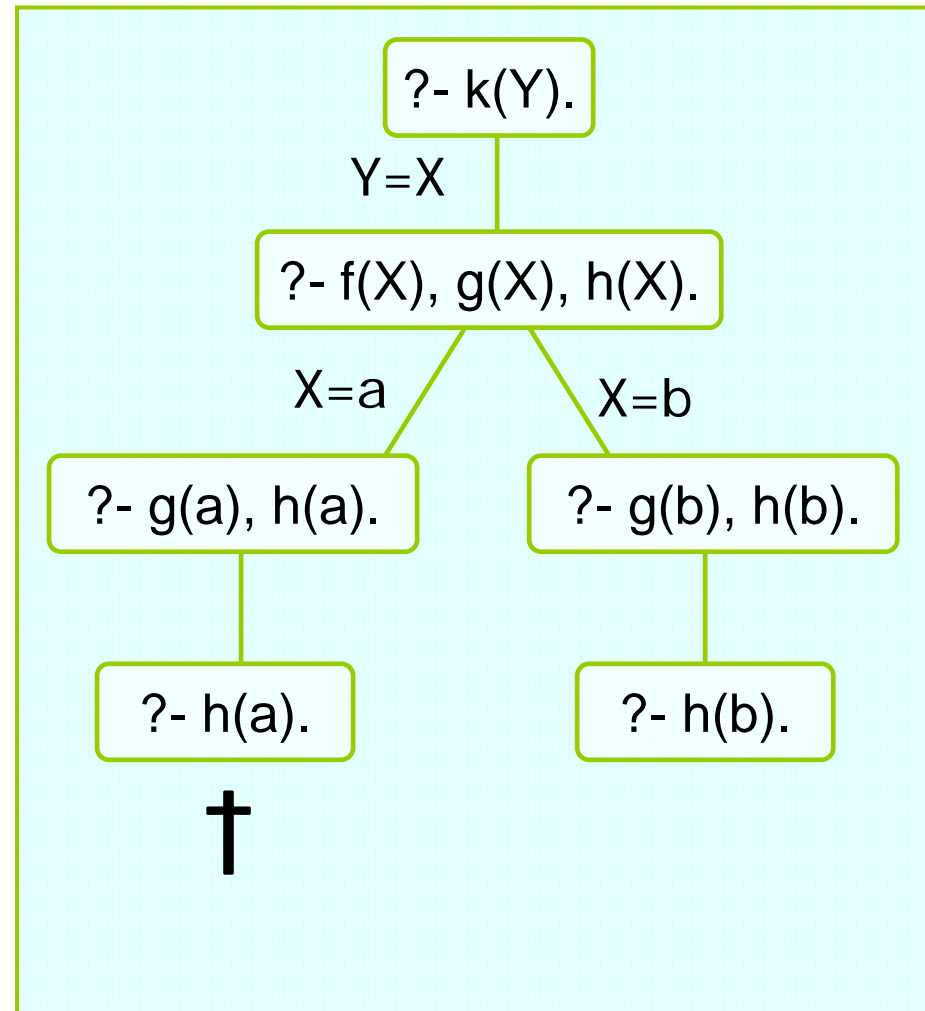
?- k(Y).



Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

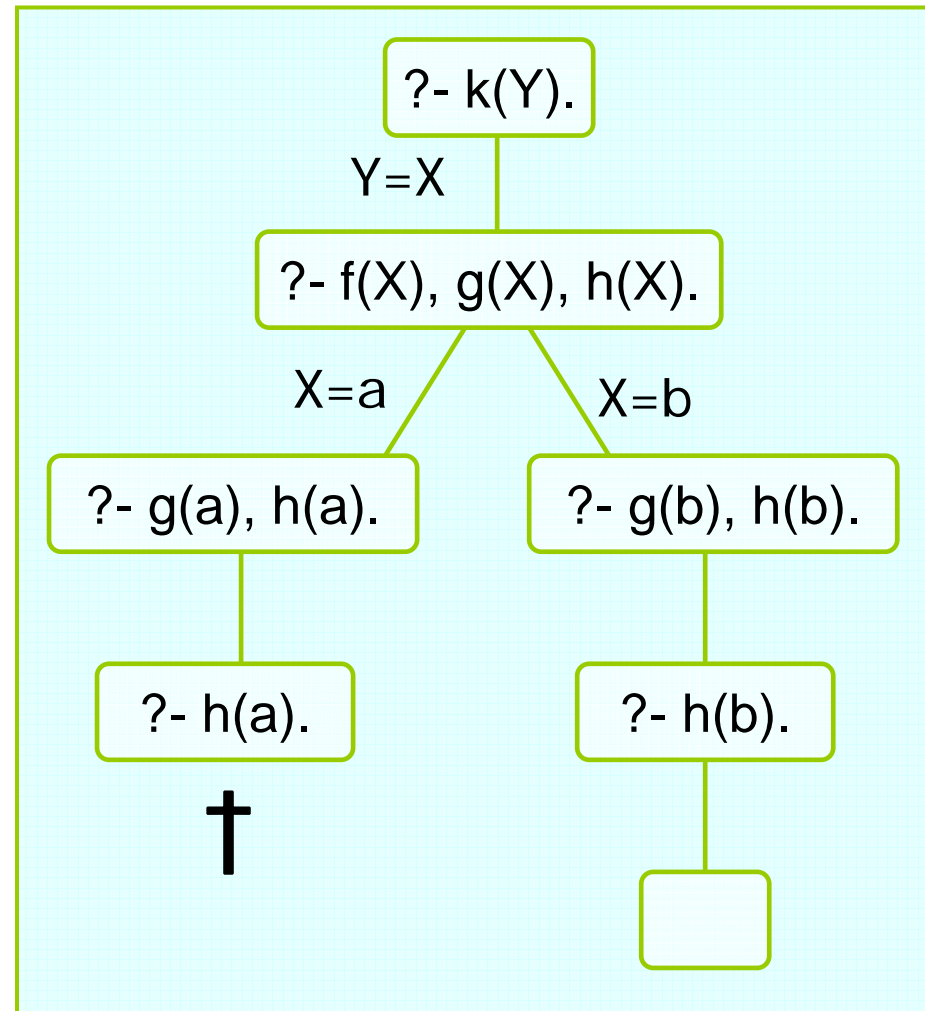
?- k(Y).



Example: search tree

f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

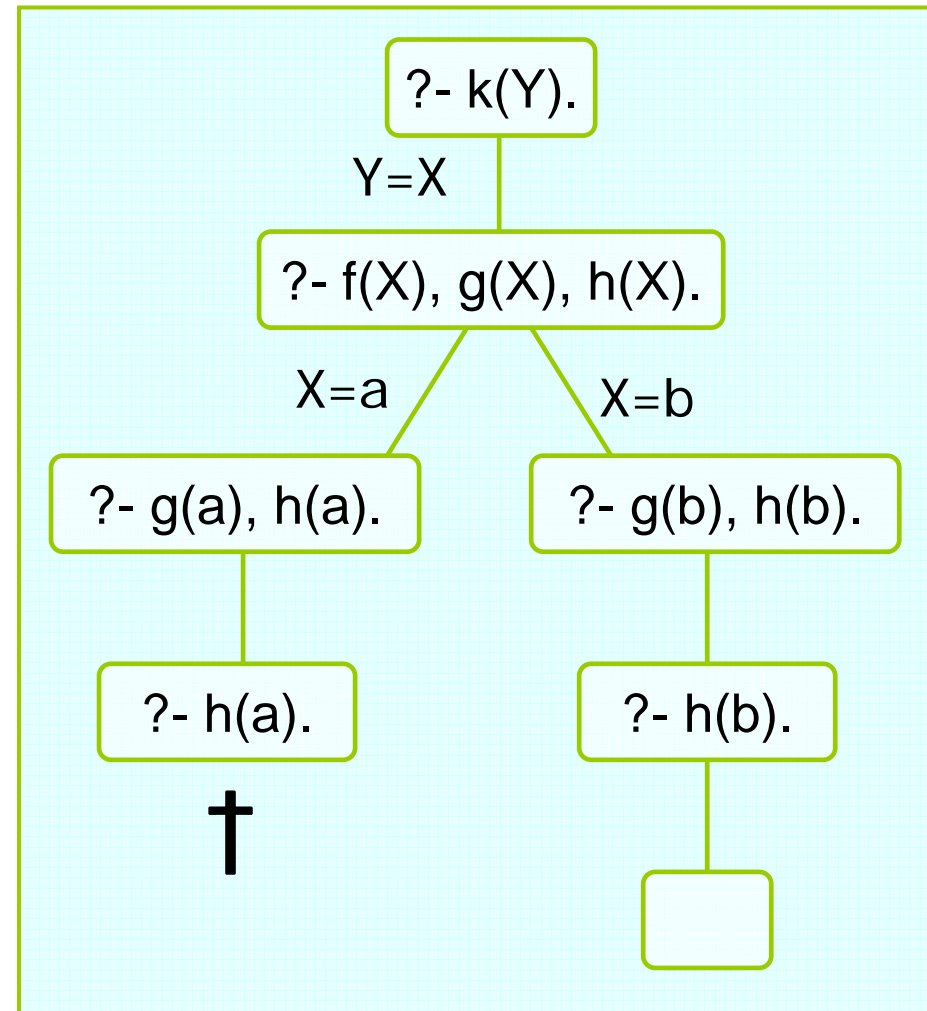
?- k(Y).
Y=b



Example: search tree

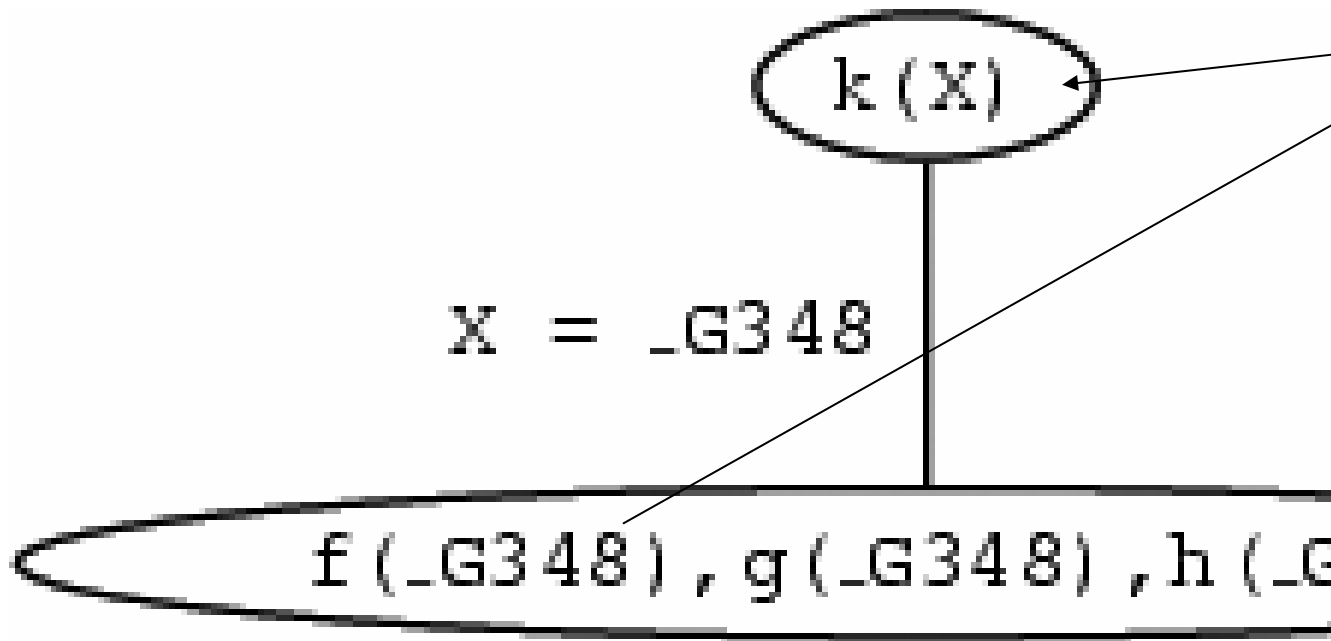
f(a).
f(b).
g(a).
g(b).
h(b).
k(X):- f(X), g(X), h(X).

?- k(Y).
Y=b;
no
?-

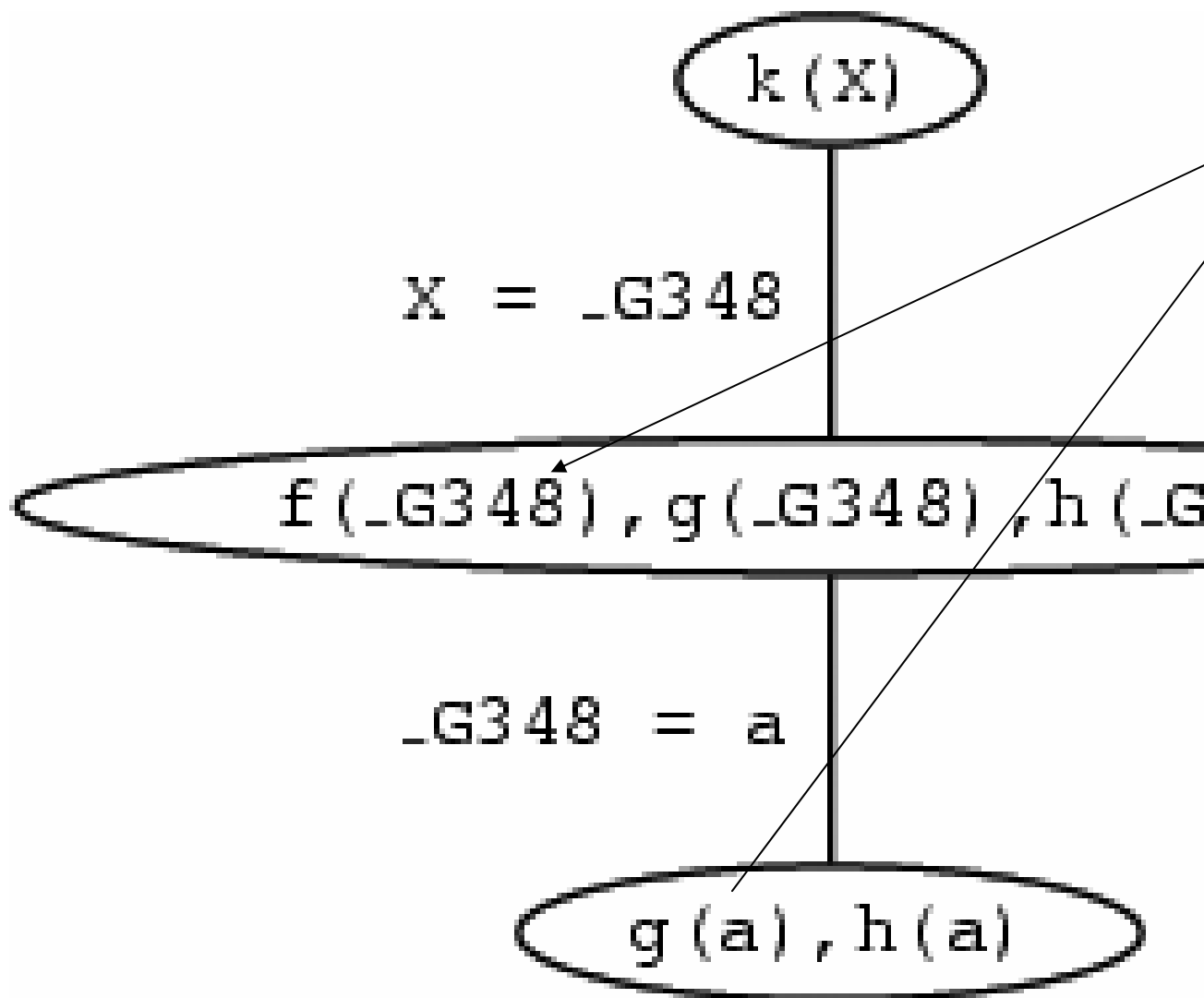


Tracing...

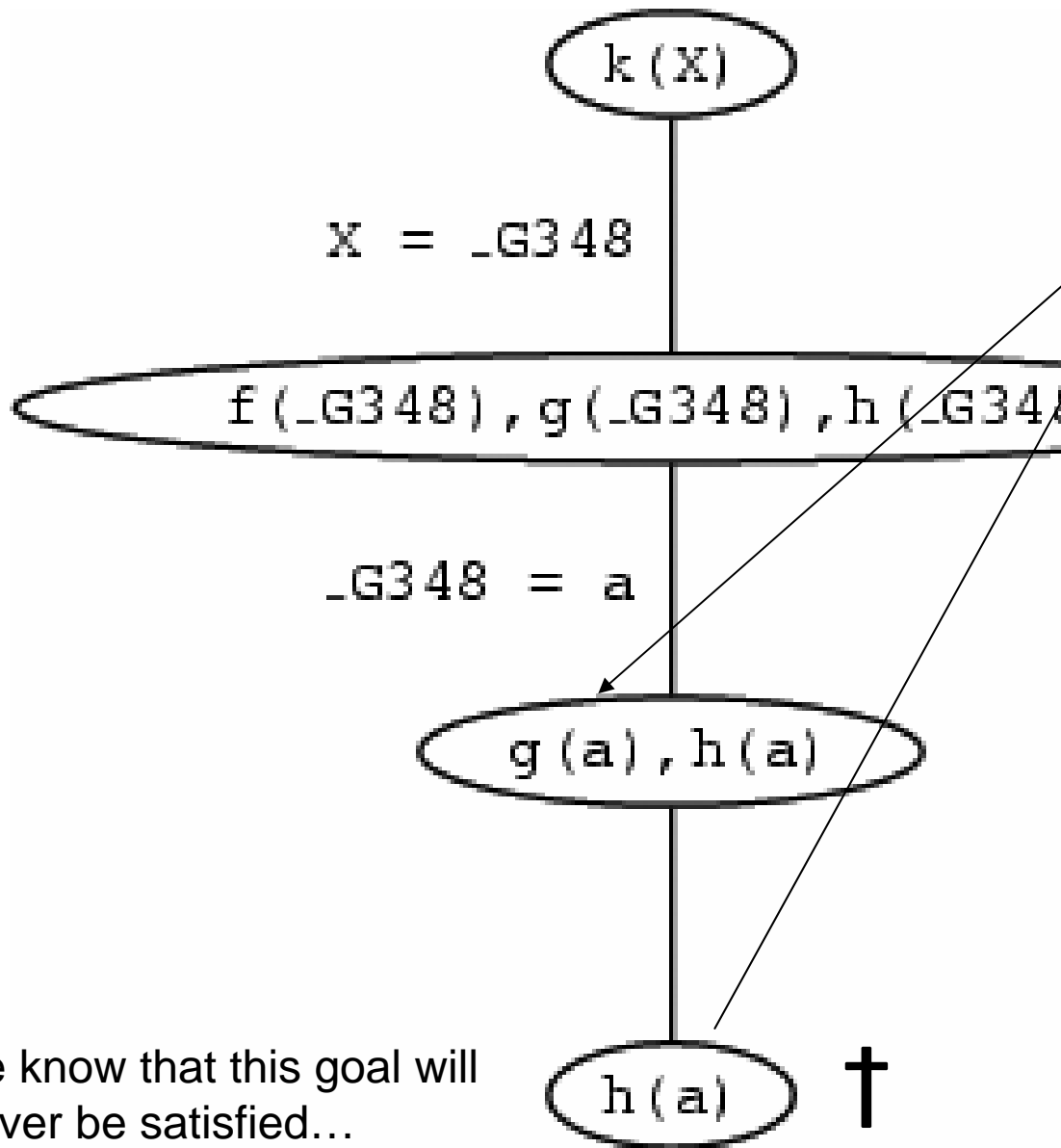
```
[debug] 13 ?- k(Y).  
T Call: (7) k(_G397)  
T Call: (8) f(_G397)  
T Exit: (8) f(a)  
T Call: (8) g(a)  
T Exit: (8) g(a)  
T Call: (8) h(a)  
T Fail: (8) h(a)  
T Redo: (8) f(_G397)  
T Exit: (8) f(b)  
T Call: (8) g(b)  
T Exit: (8) g(b)  
T Call: (8) h(b)  
T Exit: (8) h(b)  
T Exit: (7) k(b)  
Y = b.
```



[debug] 13 ?- k(Y).
T Call: (7) k(_G397)
T Call: (8) f(_G397)
T Exit: (8) f(a)
T Call: (8) g(a)
T Exit: (8) g(a)
T Call: (8) h(a)
T Fail: (8) h(a)
T Redo: (8) f(_G397)
T Exit: (8) f(b)
T Call: (8) g(b)
T Exit: (8) g(b)
T Call: (8) h(b)
T Exit: (8) h(b)
T Exit: (7) k(b)
Y = b.

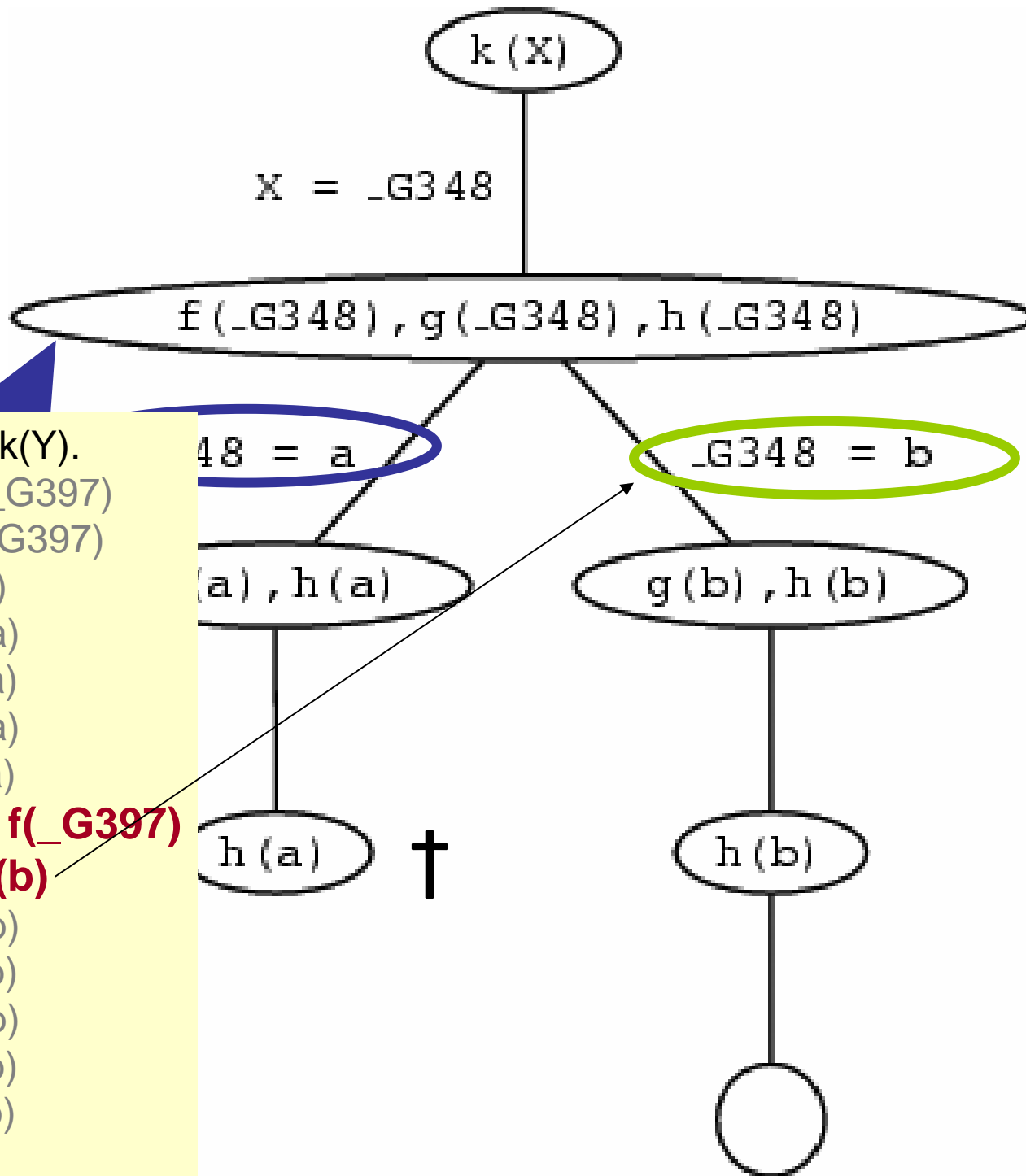


[debug] 13 ?- k(Y).
 T Call: (7) k(_G397)
 T Call: (8) f(_G397)
T Exit: (8) f(a)
 T Call: (8) g(a)
 T Exit: (8) g(a)
 T Call: (8) h(a)
 T Fail: (8) h(a)
 T Redo: (8) f(_G397)
 T Exit: (8) f(b)
 T Call: (8) g(b)
 T Exit: (8) g(b)
 T Call: (8) h(b)
 T Exit: (8) h(b)
 T Exit: (7) k(b)
 Y = b.



[debug] 13 ?- k(Y).
 T Call: (7) k(_G397)
 T Call: (8) f(_G397)
 T Exit: (8) f(a)
 T Call: (8) g(a)
T Exit: (8) g(a)
T Call: (8) h(a)
T Fail: (8) h(a)
 T Redo: (8) f(_G397)
 T Exit: (8) f(b)
 T Call: (8) g(b)
 T Exit: (8) g(b)
 T Call: (8) h(b)
 T Exit: (8) h(b)
 T Exit: (7) k(b)
 Y = b.

we know that this goal will never be satisfied...



[debug] 13 ?- k(Y).

T Call: (7) k(_G397)

T Call: (8) f(_G397)

T Exit: (8) f(a)

T Call: (8) g(a)

T Exit: (8) g(a)

T Call: (8) h(a)

T Fail: (8) h(a)

T Redo: (8) f(_G397)

T Exit: (8) f(b)

T Call: (8) g(b)

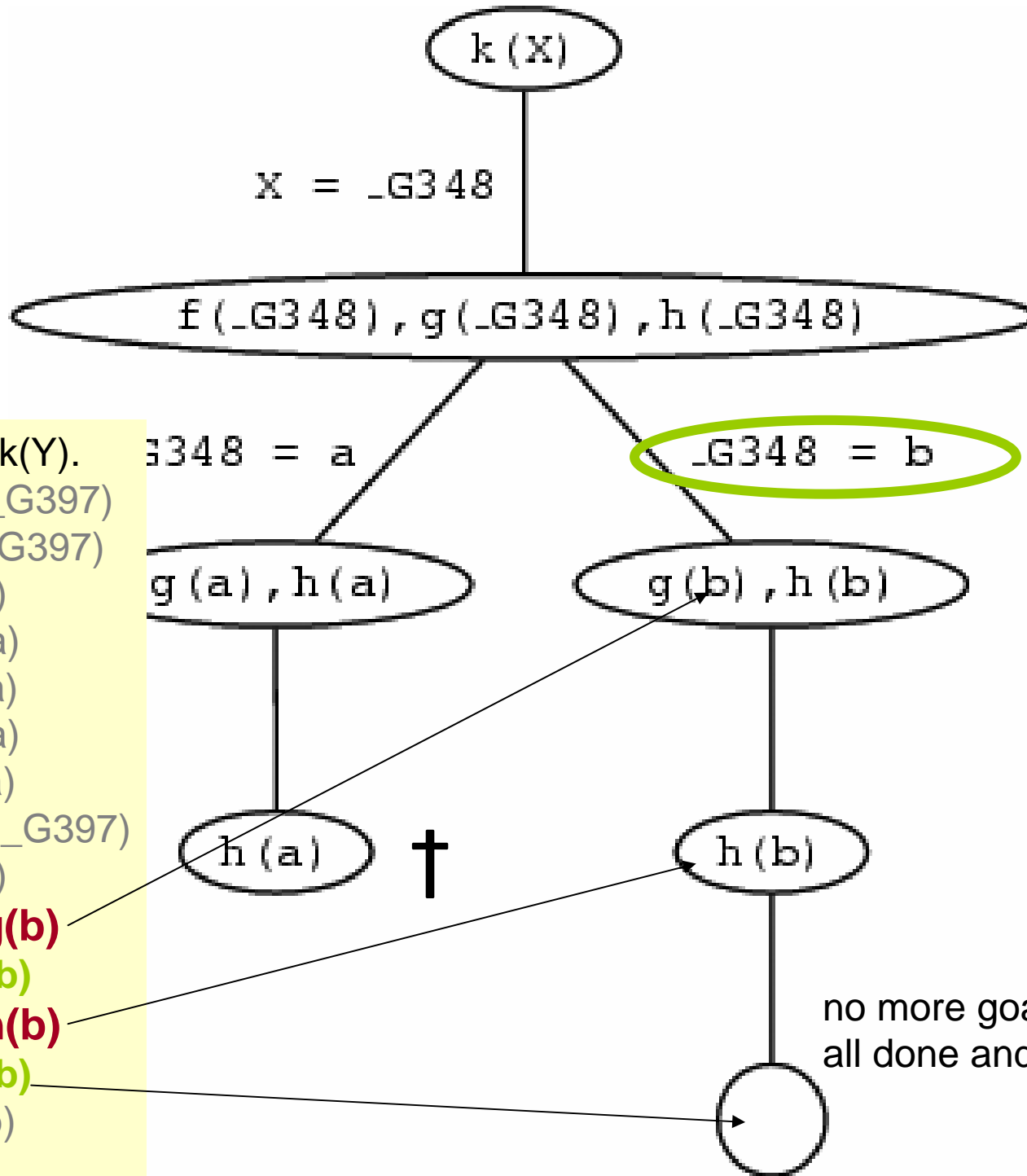
T Exit: (8) g(b)

T Call: (8) h(b)

T Exit: (8) h(b)

T Exit: (7) k(b)

Y = b.



[debug] 13 ?- k(Y).

T Call: (7) k(_G397)

T Call: (8) f(_G397)

T Exit: (8) f(a)

T Call: (8) g(a)

T Exit: (8) g(a)

T Call: (8) h(a)

T Fail: (8) h(a)

T Redo: (8) f(_G397)

T Exit: (8) f(b)

T Call: (8) g(b)

T Exit: (8) g(b)

T Call: (8) h(b)

T Exit: (8) h(b)

T Exit: (7) k(b)

Y = b.

no more goals to satisfy,
all done and happy!

What 'problem' has this funny example just solved?

```
human(X):-mortal(X).
```

f(a).
f(b).
g(a).
g(b).
h(b).

