Genesis CTT Proof Constructor
Exit
Add Lemma
Retract Lemma
Check-Context
Edit Context
Save Context
Load Context
Context file: Ex/Tarski/tarski1
Edit Query
Save Query
Load Query
Query file: Ex/Tarski/tarski.theorem
Context Representation CTT
Bracketing Normal
Scheme Vertical
S type,prop,kind
A type:kind, prop:type
R (type,type), (type,prop), (type,kind),
(prop,type), (prop,prop), (prop,kind),
(kind,type), (kind,prop), (kind,kind)
[-> == [a b:type]{x:a}b : {a b:type}type] xfy 100
[=> == [a b:prop]{assum:a}b : {a b:prop}prop] xfy 100
[t : type]
[= : {x y:t}prop] xfy 300
[<= : {x y:t}prop] xfy 400 /* leq */
[leqtrans : {x y z:t} x<=y => y<=z => x<=z] /* transitivity of <= */
[leqantisym : {x y:t} x<=y => y<=x => x=y] /* antisymmetry of <= */
[lub : {pred:{x:t}prop}t]
[upperb : {pred:{x:t}prop}{y:t}(pred y) => y<=(lub pred)]
[least : {pred:{x:t}prop}{y:t}({z:t}(pred z) => z<=y)=>((lub pred) <= y)]
[f : t->t]
[incr : {x y:t} x<=y => (f x)<=(f y)]
[po == [u:t] u<=(f u) : {u:t}prop]
[xo == (lub po) : t]
[Theorem ==
(leqantisym (f xo) xo
(upperb po (f xo)
(incr xo (f xo)
(least po (f xo)
([z:t][assum:(po z)]
(leqtrans z (f z) (f xo) assum (incr z xo (upperb po z assum)))))))
(least po (f xo)
([z:t][assum:(po z)]
(leqtrans z (f z) (f xo) assum (incr z xo (upperb po z assum))))))) :
(f xo)=xo]
Check
Smart
Analyze
Student
Alt
S-I-G
Intro
Strip
Subst
Step
Whoopy!! I found a solution all by myself!
Auto
Undo
[15] 5
20
[(leqantisym (f xo) xo (upperb po (f xo) (incr xo (f xo) (least po (f xo) ([z:t][assum:(