```
compose_append  THM

⊢  ∀T:Type, L1,L2:(T→T) List.
|  compose_list(L1) o compose_list(L2) = compose_list(L1 @ L2)
| by ...
THEN Fold `compose_list` 0
|
|\
| 1. T : Type
| 2. (T→T) List
| 3. L2 : (T→T) List
| ⊢  (λx.x) o compose_list(L2) = compose_list(L2)
|   by Ext THEN Reduce 0 THEN Auto
|
.\
. 1. T : Type
. 2. (T→T) List
. 3. u : T→T
. 4. v : (T→T) List
. 5. ∀L2:(T→T) List.
. |  compose_list(v) o compose_list(L2) = compose_list(v @ L2)
. 6. L2 : (T→T) List
. ⊢  u o compose_list(v) o compose_list(L2) = u o compose_list(v @ L2)
. | by InstHyp [L2] (-2) THENA Auto THEN RevHypSubst (-1) 0 THENA Auto
. |
. |
. 1. T : Type
. 2. (T→T) List
. 3. u : T→T
. 4. v : (T→T) List
. 5. ∀L2:(T→T) List.
.    compose_list(v) o compose_list(L2) = compose_list(v @ L2)
. 6. L2 : (T→T) List
. 7. compose_list(v) o compose_list(L2) = compose_list(v @ L2)
. ⊢  u o compose_list(v) o compose_list(L2)
.    =
.    u o compose_list(v) o compose_list(L2)
.   by Ext THEN Reduce 0 THEN Auto
END
```