```
list_all_all  THM

⊢  ∀eq:{T=₂}, P:(T→Type), L:T List.
|  ∀x∈L.P(x) ⟺ (∀x:{x:T| x(∈eq) L }. P(x))  by UnivCD
|
1. T : Type
2. eq : {T=₂}
3. P : T→Type
4. L : T List
⊢  ∀x∈L.P(x) ⟺ (∀x:{x:T| x(∈eq) L }. P(x))  by Analyze 0
|
|\
| 5. ∀x∈L.P(x)
| 6. x : {x:T| x(∈eq) L }
| ⊢  P(x)
| | by Using [‘T’,T;‘eq’,eq;‘z’,x]
| |    (FwdThru
| |     Thm* ∀eq:{T=₂}, P:(T→Prop), L:T List.
| |          ∀x∈L.P(x) ⟺ (∀z:T. z(∈eq) L ⇒ P(z))
| |     {list_all_is_member_lemma}
| |     [-2])
| |
| |\
| | .....antecedent .....
| | ⊢  x(∈eq) L  by Analyze -1 THEN Unhide
| |
| .\
| . 7. P(x)
| . ⊢  P(x)  by Trivial
|
.\
. 5. ∀x:{x:T| x(∈eq) L }. P(x)
. ⊢  ∀x∈L.P(x)  by ListInd -2 THEN AbReduce 0
. |
. |\
. | 4. T List
. | ⊢  (∀x:{x:T| False }. P(x)) ⇒ True  by Auto
. |
. .\
. . 5. u : T
. . 6. v : T List
. . 7. (∀x:{x:T| x(∈eq) v }. P(x)) ⇒ ∀x∈v.P(x)
. . ⊢  (∀x:{x:T| if eq(x,u)→ true₂ else x(∈eq) v fi }. P(x))
. . |  ⇒
. . |  P(u) & ∀x∈v.P(x)  by Auto
. . |
. . |\
. . | 8. ∀x:{x:T| if eq(x,u)→ true₂ else x(∈eq) v fi }. P(x)
. . | ⊢  P(u)  by With u (Analyze -1)
. . | |
. . | |\
. . | | .....wf .....
. . | | 7. (∀x:{x:T| x(∈eq) v }. P(x)) ⇒ ∀x∈v.P(x)
. . | | ⊢  u ∈ {x:T| if eq(x,u)→ true₂ else x(∈eq) v fi }  by Analyze
. . | | |
```

```
. . | | .....set predicate .....
. . | | ⊢  if eq(u,u)→ true₂ else u(∈eq) v fi  by IfThenElseCases 0
. . | | |
. . | | |\
. . | | | .....truecase .....
. . | | | 8. eq(u,u) = true₂
. . | | | ⊢  true₂
. . | | |   by BackThru
. . | | |      Thm* ∀P:𝔹. P ⟺ P = true₂    {assert_iff_btrue}
. . | | |
. . | | .\
. . | | . .....falsecase .....
. . | | . 8. eq(u,u) = false₂
. . | | . ⊢  u(∈eq) v  by DiscreteEq 2
. . | | . |
. . | | . 3. eq ∈ T→T→𝔹
. . | | . 4. ∀x,y:T. eq(x,y) ⟺ x = y
. . | | . 5. P : T→Type
. . | | . 6. T List
. . | | . 7. u : T
. . | | . 8. v : T List
. . | | . 9. (∀x:{x:T| x(∈eq) v }. P(x)) ⇒ ∀x∈v.P(x)
. . | | . 10. eq(u,u) = false₂
. . | | . ⊢  u(∈eq) v  by Witness4 u THEN With u (Analyze -1)
. . | | . |
. . | | . 4. P : T→Type
. . | | . 5. T List
. . | | . 6. u : T
. . | | . 7. v : T List
. . | | . 8. (∀x:{x:T| x(∈eq) v }. P(x)) ⇒ ∀x∈v.P(x)
. . | | . 9. eq(u,u) = false₂
. . | | . 10. eq(u,u) ⟺ u = u
. . | | . ⊢  u(∈eq) v  by HypSubst -2 -1 THEN AbReduce 0
. . | | . |
. . | | . 10. false₂ ⟺ u = u
. . | | . ⊢  u(∈eq) v  by AbReduce -1
. . | | . |
. . | | . 10. False ⟺ u = u
. . | | . ⊢  u(∈eq) v  by OnHyps [-1;-2;-1] Analyze
. . | | . |
. . | | . |\
. . | | . | 10. False
. . | | . | ⊢  False  by Auto
. . | | . |
. . | | . .\
. . | | . . 10. False ⇐ (u = u)
. . | | . . 11. u = u
. . | | . . ⊢  u(∈eq) v  by Analyze -2
. . | |
. . | .\
. . | . 8. P(u)
. . | . ⊢  P(u)  by Trivial
. . |
. . .\
. . . 8. ∀x:{x:T| if eq(x,u)→ true₂ else x(∈eq) v fi }. P(x)
```

```
. . . ⊢  ∀x∈v.P(x)  by HypBackchain
. . . |
. . . |\
. . . | .....wf .....
. . . | 9. x : {x:T| x(∈eq) v }
. . . | ⊢  x ∈ {x:T| if eq(x,u)→ true₂ else x(∈eq) v fi }
. . . | | by Analyze -1
. . . | |
. . . | 9. x : T
. . . | 10. x(∈eq) v
. . . | ⊢  x ∈ {x:T| if eq(x,u)→ true₂ else x(∈eq) v fi }  by Analyze
. . . | |
. . . | .....set predicate .....
. . . | ⊢  if eq(x,u)→ true₂ else x(∈eq) v fi  by IfThenElseCases 0
. . . | |
. . . | .....truecase .....
. . . | 11. eq(x,u) = true₂
. . . | ⊢  true₂
. . . |   by BackThru Thm* ∀P:𝔹. P ⟺ P = true₂    {assert_iff_btrue}
. . . |
. . . .\
. . . . .....wf .....
. . . . ⊢  {x:T| x(∈eq) v } ∈ Type  by Auto
END
```