```
list_exists_is_member_lemma  THM

⊢  ∀eq:{T=₂}, P:(T→Prop), L:T List.
|  (∃x:T. x(∈eq) L & P(x)) ⟺ ∃y∈L.P(y)
| by UnivCD THEN Analyze 0 THEN Analyze 2 THEN Analyze 0
|
|\
| 1. T : Type
| 2. eq : T→T→𝔹
| 3. (∀x,y:T. eq(x,y) ⟺ x = y) & eq ∈ T→T→𝔹
| 4. P : T→Prop
| 5. L : T List
| 6. ∃x:T. x(∈eq) L & P(x)
| ⊢  ∃y∈L.P(y)  by Analyze -1 THEN Analyze -1
| |
| 6. x : T
| 7. x(∈eq) L
| 8. P(x)
| ⊢  ∃y∈L.P(y)  by ListInd 5
| |
| |\
| | 7. P(x)
| | ⊢  x(∈eq) nil ⇒ ∃y∈nil.P(y)
| |   by Rewrite
| |      (HigherC is_member_unrollC THENC HigherC assert_evalC)
| |      0
| |
| .\
| . 7. P(x)
| . 8. u : T
| . 9. v : T List
| . 10. x(∈eq) v ⇒ ∃y∈v.P(y)
| . ⊢  x(∈eq) (u.v) ⇒ ∃y∈(u.v).P(y)  by Analyze 0
| . |
| . 11. x(∈eq) (u.v)
| . ⊢  ∃y∈(u.v).P(y)  by Rewrite (HigherC list_exists_unrollC) 0
| . |
| . ⊢  P(u) ∨ ∃y∈v.P(y)  by Decide (eq(x,u))
| . |
| . |\
| . | 12. eq(x,u)
| . | ⊢  P(u) ∨ ∃y∈v.P(y)  by DNth 1 0
| . | |
| . | ⊢  P(u)  by Analyze 3 THEN Witness3 x THEN With u (Analyze -1)
| . | |
| . | 3. eq ∈ T→T→𝔹
| . | 4. P : T→Prop
| . | 5. T List
| . | 6. x : T
| . | 7. P(x)
| . | 8. u : T
| . | 9. v : T List
| . | 10. x(∈eq) v ⇒ ∃y∈v.P(y)
| . | 11. x(∈eq) (u.v)
| . | 12. eq(x,u)
```

```
| . | 13. eq(x,u) ⟺ x = u
| . | ⊢ P(u)  by FwdThru 13 [12]
| . | |
| . | 14. x = u
| . | ⊢ P(u)  by RevHypSubst 14 0
| . |
| . .\
| . . 12. ¬eq(x,u)
| . . ⊢ P(u) ∨ ∃y∈v.P(y)  by DNth 2 0
| . . |
| . . ⊢ ∃y∈v.P(y)  by BackThru 10
| . . |
| . . ⊢ x(∈eq) v  by Rewrite (HigherC is_member_unrollC) 11
| . . |
| . . 11. if eq(x,u)→ true₂ else x(∈eq) v fi
| . . 12. ¬eq(x,u)
| . . ⊢ x(∈eq) v
| . . | by Rewrite
| . . |    (LemmaC
| . . |     Thm* ∀P:𝔹. ¬P ⟺ P = false₂    {not_assert_iff_bfalse})
| . . |    12
| . . |
| . . 12. eq(x,u) = false₂
| . . ⊢ x(∈eq) v  by IfThenElseCases 11
| . . |
| . . |\
| . . | .....truecase .....
| . . | 11. eq(x,u) = true₂
| . . | 12. true₂
| . . | 13. eq(x,u) = false₂
| . . | ⊢ x(∈eq) v  by HypSubst 11 13
| . . | |
| . . | 13. true₂ = false₂
| . . | ⊢ x(∈eq) v
| . . |   by ApFunToHypEquands ‘z’ if z→ 0 else 1 fi ℤ 13
| . . |      THEN
| . . |      AbReduce -1
| . . |
| . . .\
| . . . .....falsecase .....
| . . . 11. eq(x,u) = false₂
| . . . 12. x(∈eq) v
| . . . 13. eq(x,u) = false₂
| . . . ⊢ x(∈eq) v  by Trivial
|
.\
. 1. T : Type
. 2. eq : T→T→𝔹
. 3. (∀x,y:T. eq(x,y) ⟺ x = y) & eq ∈ T→T→𝔹
. 4. P : T→Prop
. 5. L : T List
. 6. ∃y∈L.P(y)
. ⊢ ∃x:T. x(∈eq) L & P(x)  by ListInd 5
. |
. |\
```

```
. | 5. T List
. | ⊢  ∃y∈nil.P(y) ⇒ (∃x:T. x(∈eq) nil & P(x))
. |   by Rewrite (HigherC list_exists_unrollC) 0
. |
. .\
. . 6. u : T
. . 7. v : T List
. . 8. ∃y∈v.P(y) ⇒ (∃x:T. x(∈eq) v & P(x))
. . ⊢  ∃y∈(u.v).P(y) ⇒ (∃x:T. x(∈eq) (u.v) & P(x))
. . | by Rewrite (HigherC list_exists_unrollC) 0 THEN Analyze 0
. . |
. . 9. P(u) ∨ ∃y∈v.P(y)
. . ⊢  ∃x:T. x(∈eq) (u.v) & P(x)  by Analyze 9
. . |
. . |\
. . | 9. P(u)
. . | ⊢  ∃x:T. x(∈eq) (u.v) & P(x)
. . | | by Witness u THEN Analyze 0 THEN Try Trivial
. . | |
. . | ⊢  u(∈eq) (u.v)
. . | | by Rewrite (HigherC is_member_unrollC) 0 THEN Analyze 3
. . | |
. . | 3. ∀x,y:T. eq(x,y) ⟺ x = y
. . | 4. eq ∈ T→T→𝔹
. . | 5. P : T→Prop
. . | 6. T List
. . | 7. u : T
. . | 8. v : T List
. . | 9. ∃y∈v.P(y) ⇒ (∃x:T. x(∈eq) v & P(x))
. . | 10. P(u)
. . | ⊢  if eq(u,u)→ true₂ else u(∈eq) v fi
. . | | by Witness3 u THEN With u (Analyze -1)
. . | |
. . | 3. eq ∈ T→T→𝔹
. . | 4. P : T→Prop
. . | 5. T List
. . | 6. u : T
. . | 7. v : T List
. . | 8. ∃y∈v.P(y) ⇒ (∃x:T. x(∈eq) v & P(x))
. . | 9. P(u)
. . | 10. eq(u,u) ⟺ u = u
. . | ⊢  if eq(u,u)→ true₂ else u(∈eq) v fi  by Assert (u = u)
. . | |
. . | 11. u = u
. . | ⊢  if eq(u,u)→ true₂ else u(∈eq) v fi
. . | | by Analyze 10 THEN FwdThru 11 [-1]
. . | |
. . | 10. eq(u,u) ⇒ u = u
. . | 11. (eq(u,u)) ⇐ (u = u)
. . | 12. u = u
. . | 13. eq(u,u)
. . | ⊢  if eq(u,u)→ true₂ else u(∈eq) v fi
. . | | by OnHyps [12;11;10] (λi.Thin i)
. . | |    THEN
. . | |    FwdThru Thm* ∀P:𝔹. P ⟺ P = true₂    {assert_iff_btrue}
```

```
. . | |    [-1]
. . | |
. . | 10. eq(u,u)
. . | 11. eq(u,u) = true₂
. . | ⊢  if eq(u,u)→ true₂ else u(∈eq) v fi
. . |   by HypSubst 11 0
. . |      THEN
. . |      Rewrite
. . |      (HigherC ifthenelse_evalC THENC HigherC assert_evalC)
. . |      0
. . |      THEN
. . |      Trivial
. . |
. . .\
. . . 9. ∃y∈v.P(y)
. . . ⊢  ∃x:T. x(∈eq) (u.v) & P(x)
. . . | by RenameVar ‘x’ 9 THEN Witness8 x
. . . |
. . . 8. ∃y∈v.P(y)
. . . 9. ∃x:T. x(∈eq) v & P(x)
. . . ⊢  ∃x:T. x(∈eq) (u.v) & P(x)  by Analyze 9 THEN Analyze -1
. . . |
. . . 9. x1 : T
. . . 10. x1(∈eq) v
. . . 11. P(x1)
. . . ⊢  ∃x:T. x(∈eq) (u.v) & P(x)  by Witness x1 THEN Analyze 0
. . . |
. . . |\
. . . | ⊢  x1(∈eq) (u.v)
. . . |   by Rewrite (HigherC is_member_unrollC) 0
. . . |      THEN
. . . |      GenConcl (eq(x1,u) = b)
. . . |      THEN
. . . |      BoolInd -2
. . . |      THEN
. . . |      Rewrite
. . . |      (HigherC ifthenelse_evalC

. . . |       THENC
. . . |       TryC (HigherC assert_evalC))
. . . |      0
. . . |      THEN
. . . |      Try Trivial
. . . |
. . . .\
. . . . ⊢  P(x1)  by Trivial
END
```