```
list_exists_is_member_lemma1  THM

⊢  ∀eq:{T=₂}, P:(T→Prop), x:T, L:T List.
|  x(∈eq) L ⇒ P(x) ⇒ ∃y∈L.P(y)  by RepeatFor 5 (Analyze 0)
|
1. T : Type
2. eq : {T=₂}
3. P : T→Prop
4. x : T
5. L : T List
⊢  x(∈eq) L ⇒ P(x) ⇒ ∃y∈L.P(y)  by Analyze 2
|
2. eq : T→T→𝔹
3. (∀x,y:T. eq(x,y) ⟺ x = y) & eq ∈ T→T→𝔹
4. P : T→Prop
5. x : T
6. L : T List
⊢  x(∈eq) L ⇒ P(x) ⇒ ∃y∈L.P(y)  by UnivCD
|
7. x(∈eq) L
8. P(x)
⊢  ∃y∈L.P(y)  by ListInd 6
|
|\
| 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. x : T
. | 6. T List
. | 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 HypSubst 12 11 THEN Rewrite (HigherC ifthenelse_evalC) 11
. . |
. . 11. x(∈eq) v
. . 12. eq(x,u) = false₂
. . ⊢  x(∈eq) v  by Trivial
END
```