```
not_is_member_remove_lemma  THM

⊢  ∀eq:{T=₂}, x,y:T, L:T List.
|  x(∈eq) L ⇒ ¬eq(x,y) ⇒ x(∈eq) remove(eq;y;L)  by UnivCD
|
1. T : Type
2. eq : {T=₂}
3. x : T
4. y : T
5. L : T List
6. x(∈eq) L
7. ¬eq(x,y)
⊢  x(∈eq) remove(eq;y;L)  by ListInd 5 THEN AbReduce 0
|
6. ¬eq(x,y)
7. u : T
8. v : T List
9. x(∈eq) v ⇒ x(∈eq) remove(eq;y;v)
10. if eq(x,u)→ true₂ else x(∈eq) v fi
⊢  x(∈eq) if eq(y,u)→ v else u.remove(eq;y;v) fi
| by IfThenElseCases 0
|
|\
| .....truecase .....
| 11. eq(y,u) = true₂
| ⊢  x(∈eq) v  by IfThenElseCases -2
| |
| |\
| | .....truecase .....
| | 10. eq(x,u) = true₂
| | 11. true₂
| | 12. eq(y,u) = true₂
| | ⊢  x(∈eq) v
| | | by OnHyps [12;10]
| | |    (λi.
| | |     FwdThru Thm* ∀P:𝔹. P ⟺ P = true₂    {assert_iff_btrue}
| | |     [i])
| | |
| | 13. eq(y,u)
| | 14. eq(x,u)
| | ⊢  x(∈eq) v
| | | by OnHyps [14;13]
| | |    (λi.
| | |     FwdThru
| | |     Thm* ∀T:Type, eq:{T=₂}.
| | |          (∀x,y:T. eq(x,y) ⟺ x = y) & eq ∈ T→T→𝔹
| | |     {discrete_equality_properties}
| | |     [i])
| | |
| | 15. x = u
| | 16. y = u
| | ⊢  x(∈eq) v  by Analyze 6
| | |
| | 6. u : T
| | 7. v : T List
```

```
| | 8. x(∈eq) v ⇒ x(∈eq) remove(eq;y;v)
| | 9. eq(x,u) = true₂
| | 10. true₂
| | 11. eq(y,u) = true₂
| | 12. eq(y,u)
| | 13. eq(x,u)
| | 14. x = u
| | 15. y = u
| | ⊢ eq(x,y)
| | | by BackThru
| | |    Thm* ∀T:Type, eq:{T=₂}.
| | |         (∀x,y:T. eq(x,y) ⟺ x = y) & eq ∈ T→T→𝔹
| | |    {discrete_equality_properties}
| | |
| | ⊢ x = y  by OnHyps [14;15] (λi.HypSubst i 0)
| |
| .\
| . .....falsecase .....
| . 10. eq(x,u) = false₂
| . 11. x(∈eq) v
| . 12. eq(y,u) = true₂
| . ⊢ x(∈eq) v  by Trivial
|
.\
. .....falsecase .....
. 11. eq(y,u) = false₂
. ⊢ x(∈eq) (u.remove(eq;y;v))  by AbReduce 0 THEN IfThenElseCases 0
. |
. |\
. | .....truecase .....
. | 12. eq(x,u) = true₂
. | ⊢ true₂  by Reduce 0 THEN Trivial
. |
. .\
. . .....falsecase .....
. . 12. eq(x,u) = false₂
. . ⊢ x(∈eq) remove(eq;y;v)  by HypBackchain
. . |
. . ⊢ x(∈eq) v  by HypSubst -1 -3 THEN Reduce -3 THEN Trivial
END
```