lshort_mem THM

```
⊢  ∀l:St*, eq:(St→St→𝔹).
|  (∀x,y:St. eq(x,y) ⇒ x = y) ⇒
|  (∀s:St. mem_f(St;s;l) ⟺ mem_f(St;s;LShort(l)))
| by GenRepD THEN (ListInd 2)
|
|\
| 1. St : Type
| 2. St*
| 3. eq : St→St→𝔹
| 4. ∀x,y:St. eq(x,y) ⇒ x = y
| 5. s : St
| ⊢  mem_f(St;s;nil) ⇒ mem_f(St;s;LShort(nil))
|   by (Analyze 0) THEN (RecUnfold ‘mem_f‘ -1) THEN (Reduce -1)
|
|\
| 1. St : Type
| 2. St*
| 3. eq : St→St→𝔹
| 4. ∀x,y:St. eq(x,y) ⇒ x = y
| 5. s : St
| 6. u : St
| 7. v : St*
| 8. mem_f(St;s;v) ⇒ mem_f(St;s;LShort(v))
| ⊢  mem_f(St;s;u.v) ⇒ mem_f(St;s;LShort((u.v)))
| | by RecUnfold ‘mem_f‘ 0 THEN RecUnfold ‘lshort‘ 0 THEN Reduce 0
| |
| ⊢  u = s ∨ mem_f(St;s;v) ⇒ u = s ∨ mem_f(St;s;LShort(v) \ u)
| | by (Analyze 0) THEN (Analyze -1)
| |
| |\
| | 9. u = s
| | ⊢  u = s ∨ mem_f(St;s;LShort(v) \ u)  by ProveProp
| |
| .\
| . 9. mem_f(St;s;v)
| . ⊢  u = s ∨ mem_f(St;s;LShort(v) \ u)
| . | by RWH
| . |    (LemmaC
| . |     Thm* ∀l:St*, s:St, eq:(St→St→𝔹), sx:St.
| . |          mem_f(St;sx;l \ s)
| . |          ⟺
| . |          if eq(s,sx)→ False else mem_f(St;sx;l) fi
| . |     {lremove_mem})
| . |    0
| . |
| . ⊢  u = s ∨ if eq(u,s)→ False else mem_f(St;s;LShort(v)) fi
| . | by SplitOnConclITE
| . |
| . |\
| . | .....truecase .....
| . | 10. eq(u,s)
| . | ⊢  u = s ∨ False  by (InstHyp [u;s] 4) THEN ProveProp
| . |
```

```
| . .\
| . . .....falsecase .....
| . . 10. ¬eq(u,s)
| . . ⊢  u = s ∨ mem_f(St;s;LShort(v))  by ProveProp
|
|\
| 1. St : Type
| 2. St*
| 3. eq : St→St→𝔹
| 4. ∀x,y:St. eq(x,y) ⇒ x = y
| 5. s : St
| ⊢  mem_f(St;s;LShort(nil)) ⇒ mem_f(St;s;nil)
|   by RecUnfold `lshort` 0 THEN Reduce 0
|
.\
. 1. St : Type
. 2. St*
. 3. eq : St→St→𝔹
. 4. ∀x,y:St. eq(x,y) ⇒ x = y
. 5. s : St
. 6. u : St
. 7. v : St*
. 8. mem_f(St;s;LShort(v)) ⇒ mem_f(St;s;v)
. ⊢  mem_f(St;s;LShort((u.v))) ⇒ mem_f(St;s;u.v)
. | by RecUnfold `mem_f` 0 THEN RecUnfold `lshort` 0 THEN Reduce 0
. |
. ⊢  u = s ∨ mem_f(St;s;LShort(v) \ u) ⇒ u = s ∨ mem_f(St;s;v)
. | by RWH
. |    (LemmaC
. |     Thm* ∀l:St*, s:St, eq:(St→St→𝔹), sx:St.
. |          mem_f(St;sx;l \ s)
. |          ⟺
. |          if eq(s,sx)→ False else mem_f(St;sx;l) fi
. |     {lremove_mem})
. |    0
. |
. ⊢  u = s ∨ if eq(u,s)→ False else mem_f(St;s;LShort(v)) fi ⇒
. |  u = s ∨ mem_f(St;s;v)  by SplitOnConclITE THEN (Analyze 0)
. |
. |\
. | 9. eq(u,s)
. | 10. u = s ∨ False
. | ⊢  u = s ∨ mem_f(St;s;v)  by Analyze -1 THEN Sel 1 (Analyze 0)
. |
. .\
. . 9. ¬eq(u,s)
. . 10. u = s ∨ mem_f(St;s;LShort(v))

. . ⊢  u = s ∨ mem_f(St;s;v)  by Analyze -1 THEN ProveProp
END
```