```
l_before_decomp  THM

⊢  ∀T:Type, L:T List, x,y:T.
|  x before y ∈ L ⟺ (∃A,B:T List. L = (A @ B) & (x ∈ A) & (y ∈ B))
| by Auto
|
|\
| 1. T : Type
| 2. L : T List
| 3. x : T
| 4. y : T
| 5. x before y ∈ L
| ⊢  ∃A,B:T List. L = (A @ B) & (x ∈ A) & (y ∈ B)
| | by (Unfold `l_before` -1) THEN (Lemmaize [-1])
| |    THEN
| |    InductionOnList
| |
| |\
| | 1. T : Type
| | 2. T List
| | ⊢  ∀x,y:T.
| |    [x; y] ⊆ nil
| |    ⇒
| |    (∃A,B:T List. nil = (A @ B) & (x ∈ A) & (y ∈ B))
| |   by RWO
| |      Thm* ∀x:T, L:T List. [x / L] ⊆ nil ⟺ False
| |      {cons_sublist_nil}
| |      0
| |
| .\
| . 1. T : Type
| . 2. T List
| . 3. u : T
| . 4. v : T List
| . 5. ∀x,y:T.
| . |  [x; y] ⊆ v ⇒ (∃A,B:T List. v = (A @ B) & (x ∈ A) & (y ∈ B))
| . ⊢  ∀x,y:T.
| . |  [x; y] ⊆ [u / v]
| . |  ⇒
| . |  (∃A,B:T List. [u / v] = (A @ B) & (x ∈ A) & (y ∈ B))
| . | by (RWO
| . |     Thm* ∀x1,x2:T, L1,L2:T List.
| . |          [x1 / L1] ⊆ [x2 / L2]
| . |          ⟺
| . |          x1 = x2 & L1 ⊆ L2 ∨ [x1 / L1] ⊆ L2
| . |     {cons_sublist_cons}
| . |     0)
| . |    THEN
| . |    (Analyze -1)
| . |
| . |\
| . | 1. T : Type
| . | 2. T List
| . | 3. u : T
| . | 4. v : T List
```

```
| . | 5. ∀x,y:T.
| . | |  [x; y] ⊆ v ⇒ (∃A,B:T List. v = (A @ B) & (x ∈ A) & (y ∈ B))
| . | 6. x : T
| . | 7. y : T
| . | 8. x = u & [y] ⊆ v
| . | ⊢  ∃A,B:T List. [u / v] = (A @ B) & (x ∈ A) & (y ∈ B)
| . | | by (InstConcl [[u];v]) THEN (Reduce 0)
| . | |
| . | |\
| . | | 1. T : Type
| . | | 2. T List
| . | | 3. u : T
| . | | 4. v : T List
| . | | 5. ∀x,y:T.
| . | |    [x; y] ⊆ v
| . | |    ⇒
| . | |    (∃A,B:T List. v = (A @ B) & (x ∈ A) & (y ∈ B))
| . | | 6. x : T
| . | | 7. y : T
| . | | 8. x = u
| . | | 9. [y] ⊆ v
| . | | ⊢  (x ∈ [u])
| . | |   by RW0 Thm* ∀a,x:T. (x ∈ [a]) ⟺ x = a    {member_singleton}
| . | |     0
| . | |
| . | .\
| . | . 1. T : Type
| . | . 2. T List
| . | . 3. u : T
| . | . 4. v : T List
| . | . 5. ∀x,y:T.
| . | .    [x; y] ⊆ v
| . | .    ⇒
| . | .    (∃A,B:T List. v = (A @ B) & (x ∈ A) & (y ∈ B))
| . | . 6. x : T
| . | . 7. y : T
| . | . 8. x = u
| . | . 9. [y] ⊆ v
| . | . ⊢  (y ∈ v)
| . | .   by RW0
| . | .      Thm* ∀x:T, L:T List. (x ∈ L) ⟺ [x] ⊆ L
| . | .      {member_iff_sublist}
| . | .      0
| . |
| . .\
| . . 1. T : Type
| . . 2. T List
| . . 3. u : T
| . . 4. v : T List
| . . 5. ∀x,y:T.
| . . |  [x; y] ⊆ v ⇒ (∃A,B:T List. v = (A @ B) & (x ∈ A) & (y ∈ B))
| . . 6. x : T
| . . 7. y : T
| . . 8. [x; y] ⊆ v
| . . ⊢  ∃A,B:T List. [u / v] = (A @ B) & (x ∈ A) & (y ∈ B)
```

```
| . . | by (InstHyp [x;y] 5) THEN ExRepD THEN (InstConcl [[u / A];B])
| . . |    THEN
| . . |    (Reduce 0)
| . . |
| . . |\
| . . | 1. T : Type
| . . | 2. T List
| . . | 3. u : T
| . . | 4. v : T List
| . . | 5. ∀x,y:T.
| . . |    [x; y] ⊆ v
| . . |    ⇒
| . . |    (∃A,B:T List. v = (A @ B) & (x ∈ A) & (y ∈ B))
| . . | 6. x : T
| . . | 7. y : T
| . . | 8. [x; y] ⊆ v
| . . | 9. A : T List
| . . | 10. B : T List
| . . | 11. v = (A @ B)
| . . | 12. (x ∈ A)
| . . | 13. (y ∈ B)
| . . | ⊢  [u / v] = [u / (A @ B)]  by Analyze
| . . |
| . . .\
| . . . 1. T : Type
| . . . 2. T List
| . . . 3. u : T
| . . . 4. v : T List
| . . . 5. ∀x,y:T.
| . . .    [x; y] ⊆ v
| . . .    ⇒
| . . .    (∃A,B:T List. v = (A @ B) & (x ∈ A) & (y ∈ B))
| . . . 6. x : T
| . . . 7. y : T
| . . . 8. [x; y] ⊆ v
| . . . 9. A : T List
| . . . 10. B : T List
| . . . 11. v = (A @ B)
| . . . 12. (x ∈ A)
| . . . 13. (y ∈ B)
| . . . ⊢  (x ∈ [u / A])  by ObviousFrom [-2]
|
.\
. 1. T : Type
. 2. L : T List
. 3. x : T
. 4. y : T
. 5. ∃A,B:T List. L = (A @ B) & (x ∈ A) & (y ∈ B)
. ⊢  x before y ∈ L
.   by (ExRepD THEN (SubstFor L 0)
.       THEN
.       (RWO
.        Thm* ∀A,B:T List, x,y:T.
.             x before y ∈ A @ B
.             ⟺
```

```
.             x before y ∈ A ∨ x before y ∈ B ∨ (x ∈ A) & (y ∈ B)
.        {l_before_append_iff}
.        0)
.       THEN
.       OrRight)
.      THEN
.      OrRight
END
```