```
sublist_append_iff  THM

⊢  ∀T:Type, C,A,B:T List.
|  C ⊆ A @ B ⟺ (∃A',B':T List. C = (A' @ B') & A' ⊆ A & B' ⊆ B)
| by InductionOnList THEN InductionOnList THEN (Reduce 0)
|    THEN
|    (Try (RWO Thm* ∀L:T List. nil ⊆ L ⟺ True    {nil_sublist} 0))
|    THEN
|    (Try
|     (Auto THEN (InstConcl [nil;nil]) THEN (Reduce 0)
|      THEN
|      (RWO Thm* ∀L:T List. nil ⊆ L ⟺ True    {nil_sublist} 0)))
|    THEN
|    (Try
|     ((RWO Thm* ∀L:T List. L ⊆ nil ⟺ L = nil    {sublist_nil} 0)
|      THEN
|      ObviousConcl))
|
1. T : Type
2. T List
3. u : T
4. v : T List
5. ∀A,B:T List.
|  v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
6. T List
7. u1 : T
8. v1 : T List
9. ∀B:T List.
|  [u / v] ⊆ v1 @ B
|  ⟺
|  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
⊢  ∀B:T List.
|  [u / v] ⊆ [u1 / (v1 @ B)]
|  ⟺
|  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ [u1 / v1] & B' ⊆ 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
|
|\
| 1. T : Type
| 2. T List
| 3. u : T
| 4. v : T List
| 5. ∀A,B:T List.
| |  v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
| 6. T List
| 7. u1 : T
| 8. v1 : T List
| 9. ∀B:T List.
| |  [u / v] ⊆ v1 @ B
| |  ⟺
| |  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
```

```
| 10. B : T List
| 11. u = u1 & v ⊆ v1 @ B ∨ [u / v] ⊆ v1 @ B
| ⊢  ∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ [u1 / v1] & B' ⊆ B
| | by Analyze -1
| |
| |\
| | 1. T : Type
| | 2. T List
| | 3. u : T
| | 4. v : T List
| | 5. ∀A,B:T List.
| | |  v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
| | 6. T List
| | 7. u1 : T
| | 8. v1 : T List
| | 9. ∀B:T List.
| | |  [u / v] ⊆ v1 @ B
| | |  ⟺
| | |  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
| | 10. B : T List
| | 11. u = u1
| | 12. v ⊆ v1 @ B
| | ⊢  ∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ [u1 / v1] & B' ⊆ B
| | | by (RW (HypC 5) -1) THEN ExRepD THEN (InstConcl [[u / A'];B'])
| | | |    THEN
| | | |    (Reduce 0)
| | | |
| | | |\
| | | | 1. T : Type
| | | | 2. T List
| | | | 3. u : T
| | | | 4. v : T List
| | | | 5. ∀A,B:T List.
| | | |    v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
| | | | 6. T List
| | | | 7. u1 : T
| | | | 8. v1 : T List
| | | | 9. ∀B:T List.
| | | |    [u / v] ⊆ v1 @ B
| | | |    ⟺
| | | |    (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
| | | | 10. B : T List
| | | | 11. u = u1
| | | | 12. A' : T List
| | | | 13. B' : T List
| | | | 14. v = (A' @ B')
| | | | 15. A' ⊆ v1
| | | | 16. B' ⊆ B
| | | | ⊢  [u / v] = [u / (A' @ B')]  by Analyze
| | | |
| | | .\
| | | . 1. T : Type
| | | . 2. T List
| | | . 3. u : T
| | | . 4. v : T List
```

```
| | . 5. ∀A,B:T List.
| | .    v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
| | . 6. T List
| | . 7. u1 : T
| | . 8. v1 : T List
| | . 9. ∀B:T List.
| | .    [u / v] ⊆ v1 @ B
| | .    ⟺
| | .    (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
| | . 10. B : T List
| | . 11. u = u1
| | . 12. A' : T List
| | . 13. B' : T List
| | . 14. v = (A' @ B')
| | . 15. A' ⊆ v1
| | . 16. B' ⊆ B
| | . ⊢  [u / A'] ⊆ [u1 / v1]
| | .   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
| | .      OrLeft
| |
| .\
| . 1. T : Type
| . 2. T List
| . 3. u : T
| . 4. v : T List
| . 5. ∀A,B:T List.
| . |  v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
| . 6. T List
| . 7. u1 : T
| . 8. v1 : T List
| . 9. ∀B:T List.
| . |  [u / v] ⊆ v1 @ B
| . |  ⟺
| . |  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
| . 10. B : T List
| . 11. [u / v] ⊆ v1 @ B
| . ⊢  ∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ [u1 / v1] & B' ⊆ B
| . | by (RW (HypC -3) -1) THEN (RepeatFor 4 (ParallelOp -1))
| . |
| . 1. T : Type
| . 2. T List
| . 3. u : T
| . 4. v : T List
| . 5. ∀A,B:T List.
| .    v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
| . 6. T List
| . 7. u1 : T
| . 8. v1 : T List
```

```
| . 9. ∀B:T List.
| .    [u / v] ⊆ v1 @ B
| .    ⟺
| .    (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
| . 10. T List
| . 11. A' : T List
| . 12. T List
| . 13. A' ⊆ v1
| . ⊢  A' ⊆ [u1 / v1]  by (Analyze -3) THEN (ObviousFrom [-1])
|
.\
. 1. T : Type
. 2. T List
. 3. u : T
. 4. v : T List
. 5. ∀A,B:T List.
. |  v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. 6. T List
. 7. u1 : T
. 8. v1 : T List
. 9. ∀B:T List.
. |  [u / v] ⊆ v1 @ B
. |  ⟺
. |  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. 10. B : T List
. 11. ∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ [u1 / v1] & B' ⊆ B
. ⊢  u = u1 & v ⊆ v1 @ B ∨ [u / v] ⊆ v1 @ B
. | by ExRepD THEN (Analyze -5) THEN (All Reduce)
. |
. |\
. | 1. T : Type
. | 2. T List
. | 3. u : T
. | 4. v : T List
. | 5. ∀A,B:T List.
. |    v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. | 6. T List
. | 7. u1 : T
. | 8. v1 : T List
. | 9. ∀B:T List.
. |    [u / v] ⊆ v1 @ B
. |    ⟺
. |    (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. | 10. B : T List
. | 11. B' : T List
. | 12. [u / v] = B'
. | 13. nil ⊆ [u1 / v1]
. | 14. B' ⊆ B
. | ⊢  u = u1 & v ⊆ v1 @ B ∨ [u / v] ⊆ v1 @ B
. |   by OrRight THEN (RW (HypC 9) 0) THEN (InstConcl [nil;B'])
. |      THEN
. |      (Reduce 0)
. |      THEN
. |      ObviousConcl
. |
```

```
. .\
. . 1. T : Type
. . 2. T List
. . 3. u : T
. . 4. v : T List
. . 5. ∀A,B:T List.
. . |  v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. . 6. T List
. . 7. u1 : T
. . 8. v1 : T List
. . 9. ∀B:T List.
. . |  [u / v] ⊆ v1 @ B
. . |  ⟺
. . |  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. . 10. B : T List
. . 11. u2 : T
. . 12. v2 : T List
. . 13. B' : T List
. . 14. [u / v] = [u2 / (v2 @ B')]
. . 15. [u2 / v2] ⊆ [u1 / v1]
. . 16. B' ⊆ B
. . ⊢  u = u1 & v ⊆ v1 @ B ∨ [u / v] ⊆ v1 @ 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}
. . |     -2)
. . |    THEN
. . |    (Analyze -2)
. . |
. . |\
. . | 1. T : Type
. . | 2. T List
. . | 3. u : T
. . | 4. v : T List
. . | 5. ∀A,B:T List.
. . | |  v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. . | 6. T List
. . | 7. u1 : T
. . | 8. v1 : T List
. . | 9. ∀B:T List.
. . | |  [u / v] ⊆ v1 @ B
. . | |  ⟺
. . | |  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. . | 10. B : T List
. . | 11. u2 : T
. . | 12. v2 : T List
. . | 13. B' : T List
. . | 14. [u / v] = [u2 / (v2 @ B')]
. . | 15. u2 = u1 & v2 ⊆ v1
. . | 16. B' ⊆ B
. . | ⊢  u = u1 & v ⊆ v1 @ B ∨ [u / v] ⊆ v1 @ B  by OrLeft
. . | |
```

```
. . | |\
. . | | 1. T : Type
. . | | 2. T List
. . | | 3. u : T
. . | | 4. v : T List
. . | | 5. ∀A,B:T List.
. . | |    v ⊆ A @ B
. . | |    ⟺
. . | |    (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. . | | 6. T List
. . | | 7. u1 : T
. . | | 8. v1 : T List
. . | | 9. ∀B:T List.
. . | |    [u / v] ⊆ v1 @ B
. . | |    ⟺
. . | |    (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. . | | 10. B : T List
. . | | 11. u2 : T
. . | | 12. v2 : T List
. . | | 13. B' : T List
. . | | 14. [u / v] = [u2 / (v2 @ B')]
. . | | 15. u2 = u1
. . | | 16. v2 ⊆ v1
. . | | 17. B' ⊆ B
. . | | ⊢  u = u1  by ObviousFrom [-4;-3]
. . | |
. . | .\
. . | . 1. T : Type
. . | . 2. T List
. . | . 3. u : T
. . | . 4. v : T List
. . | . 5. ∀A,B:T List.
. . | . |  v ⊆ A @ B
. . | . |  ⟺
. . | . |  (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. . | . 6. T List
. . | . 7. u1 : T
. . | . 8. v1 : T List
. . | . 9. ∀B:T List.
. . | . |  [u / v] ⊆ v1 @ B
. . | . |  ⟺
. . | . |  (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. . | . 10. B : T List
. . | . 11. u2 : T
. . | . 12. v2 : T List
. . | . 13. B' : T List
. . | . 14. [u / v] = [u2 / (v2 @ B')]
. . | . 15. u2 = u1
. . | . 16. v2 ⊆ v1
. . | . 17. B' ⊆ B
. . | . ⊢  v ⊆ v1 @ B  by (RW (HypC 5) 0) THEN (InstConcl [v2;B'])
. . | . |
. . | . 1. T : Type
. . | . 2. T List
. . | . 3. u : T
```

```
. . | . 4. v : T List
. . | . 5. ∀A,B:T List.
. . | .    v ⊆ A @ B
. . | .    ⟺
. . | .    (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. . | . 6. T List
. . | . 7. u1 : T
. . | . 8. v1 : T List
. . | . 9. ∀B:T List.
. . | .    [u / v] ⊆ v1 @ B
. . | .    ⟺
. . | .    (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. . | . 10. B : T List
. . | . 11. u2 : T
. . | . 12. v2 : T List
. . | . 13. B' : T List
. . | . 14. [u / v] = [u2 / (v2 @ B')]
. . | . 15. u2 = u1
. . | . 16. v2 ⊆ v1
. . | . 17. B' ⊆ B
. . | . ⊢  v = (v2 @ B')  by ObviousFrom [-4]
. . |
. . .\
. . . 1. T : Type
. . . 2. T List
. . . 3. u : T
. . . 4. v : T List
. . . 5. ∀A,B:T List.
. . .    v ⊆ A @ B ⟺ (∃A',B':T List. v = (A' @ B') & A' ⊆ A & B' ⊆ B)
. . . 6. T List
. . . 7. u1 : T
. . . 8. v1 : T List
. . . 9. ∀B:T List.
. . .    [u / v] ⊆ v1 @ B
. . .    ⟺
. . .    (∃A',B':T List. [u / v] = (A' @ B') & A' ⊆ v1 & B' ⊆ B)
. . . 10. B : T List
. . . 11. u2 : T
. . . 12. v2 : T List
. . . 13. B' : T List
. . . 14. [u / v] = [u2 / (v2 @ B')]
. . . 15. [u2 / v2] ⊆ v1
. . . 16. B' ⊆ B
. . . ⊢  u = u1 & v ⊆ v1 @ B ∨ [u / v] ⊆ v1 @ B
. . .   by OrRight THEN (RW (HypC 9) 0)
. . .      THEN
. . .      (InstConcl [[u2 / v2];B'])
. . .      THEN
. . .      (Reduce 0)
END
```