```
equal_appends  THM

⊢  ∀T:Type, a,c,b,d:T List.
|  (a @ b) = (c @ d)
|  ⇒
|  (∃e:T List. a = (c @ e) & d = (e @ b) ∨ c = (a @ e) & b = (e @ d))
| by InductionOnList THEN (Reduce 0)
|
|\
| 1. T : Type
| 2. T List
| ⊢  ∀c,b,d:T List.
|    b = (c @ d)
|    ⇒
|    (∃e:T List. nil = (c @ e) & d = (e @ b) ∨ c = e & b = (e @ d))
|   by Obvious
|
.\
. 1. T : Type
. 2. T List
. 3. u : T
. 4. v : T List
. 5. ∀c,b,d:T List.
. |  (v @ b) = (c @ d)
. |  ⇒
. |  (∃e:T List.
. |   v = (c @ e) & d = (e @ b) ∨ c = (v @ e) & b = (e @ d))
. ⊢  ∀c,b,d:T List.
. |  [u / (v @ b)] = (c @ d)
. |  ⇒
. |  (∃e:T List.
. |   [u / v] = (c @ e) & d = (e @ b)
. |   ∨ c = [u / (v @ e)] & b = (e @ d))
. | by InductionOnList THEN (Reduce 0)
. |
. |\
. | 1. T : Type
. | 2. T List
. | 3. u : T
. | 4. v : T List
. | 5. ∀c,b,d:T List.
. |    (v @ b) = (c @ d)
. |    ⇒
. |    (∃e:T List.
. |     v = (c @ e) & d = (e @ b) ∨ c = (v @ e) & b = (e @ d))
. | 6. T List
. | ⊢  ∀b,d:T List.
. |    [u / (v @ b)] = d
. |    ⇒
. |    (∃e:T List.
. |     [u / v] = e & d = (e @ b) ∨ nil = [u / (v @ e)] & b = (e @ d))
. |   by Obvious
. |
. .\
. . 1. T : Type
```

```
. . 2. T List
. . 3. u : T
. . 4. v : T List
. . 5. ∀c,b,d:T List.
. .    (v @ b) = (c @ d)
. .    ⇒
. .    (∃e:T List.
. .     v = (c @ e) & d = (e @ b) ∨ c = (v @ e) & b = (e @ d))
. . 6. T List
. . 7. u1 : T
. . 8. v1 : T List
. . 9. ∀b,d:T List.
. .    [u / (v @ b)] = (v1 @ d)
. .    ⇒
. .    (∃e:T List.
. .     [u / v] = (v1 @ e) & d = (e @ b)
. .     ∨ v1 = [u / (v @ e)] & b = (e @ d))
. . ⊢  ∀b,d:T List.
. .    [u / (v @ b)] = [u1 / (v1 @ d)]
. .    ⇒
. .    (∃e:T List.
. .     [u / v] = [u1 / (v1 @ e)] & d = (e @ b)
. .     ∨ [u1 / v1] = [u / (v @ e)] & b = (e @ d))
. .   by Auto THEN (SplitCons -1) THEN (InstHyp [v1;b;d] 5)
. .      THEN
. .      (ParallelOp -1)
. .      THEN
. .      (ParallelOp -1)
. .      THEN
. .      (ObviousFrom [-1;-4])
END
```