```
equal_appends_eq  THM

⊢  ∀T:Type, x1,z,x2,x3:T List.
|  ||x1|| = ||z|| ⇒ (x1 @ x2) = (z @ x3) ⇒ x1 = z & x2 = x3
| by Auto
|    THEN
|    (FwdThru
|     Thm* ∀x1,z,x2,x3:T List.
|          ||x1||≤||z||
|          ⇒
|          (x1 @ x2) = (z @ x3)
|          ⇒
|          (∃z':T List. z = (x1 @ z') & x2 = (z' @ x3))
|     {equal_appends_case1}
|     [-1])
|    THEN
|    ExRepD
|
1. T : Type
2. x1 : T List
3. z : T List
4. x2 : T List
5. x3 : T List
6. ||x1|| = ||z||
7. (x1 @ x2) = (z @ x3)
8. z' : T List
9. z = (x1 @ z')
10. x2 = (z' @ x3)
⊢  x1 = z & x2 = x3  by Analyze -3
|
|\
| 1. T : Type
| 2. x1 : T List
| 3. z : T List
| 4. x2 : T List
| 5. x3 : T List
| 6. ||x1|| = ||z||
| 7. (x1 @ x2) = (z @ x3)
| 8. z = (x1 @ nil)
| 9. x2 = (nil @ x3)
| ⊢  x1 = z & x2 = x3  by All Reduce
| |
| 1. T : Type
| 2. x1 : T List
| 3. z : T List
| 4. x2 : T List
| 5. x3 : T List
| 6. ||x1|| = ||z||
| 7. (x1 @ x2) = (z @ x3)
| 8. z = (x1 @ nil)
| 9. x2 = x3
| ⊢  x1 = z
|   by RWO Thm* ∀l:T List. (l @ nil) ∼ l    {append_nil_sq} -2
|
.\
```

```
. 1. T : Type
. 2. x1 : T List
. 3. z : T List
. 4. x2 : T List
. 5. x3 : T List
. 6. ||x1|| = ||z||
. 7. (x1 @ x2) = (z @ x3)
. 8. u : T
. 9. v : T List
. 10. z = (x1 @ [u / v])
. 11. x2 = ([u / v] @ x3)
. ⊢  x1 = z & x2 = x3
.   by (MoveToConcl 6) THEN (SubstFor z 0)
.      THEN
.      (RWO
.       Thm* ∀as,bs:T List. ||as @ bs|| = ||as||+||bs|| ∈ ℤ
.       {length_append}
.       0)
END
```