```
mn_23_lem_1  THM

⊢  ∀R:(Alph*→Alph*→Prop).
|  Fin(Alph) ⇒
|  (EquivRel x,y:Alph*. x R y) ⇒
|  Fin(x,y:Alph*//(x R y)) ⇒
|  (∀x,y,z:Alph*. (x R y) ⇒ ((z @ x) R (z @ y))) ⇒
|  (∀g:((x,y:Alph*//(x R y))→𝔹), x,y:x,y:Alph*//(x R y). Dec(x Rg y))
| by RepD
|
1. Alph : Type
2. R : Alph*→Alph*→Prop
3. Fin(Alph)
4. EquivRel x,y:Alph*. x R y
5. Fin(x,y:Alph*//(x R y))
6. ∀x,y,z:Alph*. (x R y) ⇒ ((z @ x) R (z @ y))
7. g : (x,y:Alph*//(x R y))→𝔹
8. x : x,y:Alph*//(x R y)
9. y : x,y:Alph*//(x R y)
⊢  Dec(x Rg y)  by Unfold ‘lquo_rel‘ 0 THEN Reduce 0
|
⊢  Dec(∀z:Alph*. g(z@q x) ⟺ g(z@q y))
| by RWH (LemmaC Thm* ∀a,b:𝔹. (a ⟺ b) ⟺ a = b    {assert_iff_eq}) 0
|
⊢  Dec(∀z:Alph*. g(z@q x) = g(z@q y))
| by Inst
|    Thm* ∀P:(T→Prop).
|         (∀x:T. Dec(P(x))) & Dec(∃x:T. ¬P(x)) ⇒ Dec(∀x:T. P(x))
|    {auto2_lemma_0}
|    [Alph*;λz.g(z@q x) = g(z@q y)]
|
|\
| .....decidable? .....
| 10. x@0 : Alph*
| ⊢  Dec((λz.g(z@q x) = g(z@q y))(x@0))  by Reduce 0
| |
| ⊢  Dec(g(x@0@q x) = g(x@0@q y))  by ProveDecidable
|
|\
| .....decidable? .....
| ⊢  Dec(∃x@0:Alph*. ¬(λz.g(z@q x) = g(z@q y))(x@0))  by Reduce 0
| |
| ⊢  Dec(∃x@0:Alph*. ¬g(x@0@q x) = g(x@0@q y))
| | by RWH
| |    (RevLemmaC Thm* ∀p,q:𝔹. p =b q = false2 ⟺ ¬p = q    {beq_neq})
| |    0
| |
| ⊢  Dec(∃x@0:Alph*. (g(x@0@q x)) =b (g(x@0@q y)) = false2)
| | by Assert
| |    (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| |     ,λa,p. p/x,y.<a.x,a.y>>
| |     ∈ ActionSet(Alph))
| |
| |\
| | .....assertion .....
```

```
| | ⊢  <(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| | |  ,λa,p. p/x,y.<a.x,a.y>>
| | |  ∈ ActionSet(Alph)  by Unfold ‘action_set‘ 0
| | |
| | |\
| | | 10. a : Alph
| | | 11. p : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| | | 12. x1 : x,y:Alph*//(x R y)
| | | 13. y1 : x,y:Alph*//(x R y)
| | | 14. p = <x1,y1>
| | | ⊢  a.x1 ∈ x,y:Alph*//(x R y)  by (Analyze 12) THEN Analyze
| | | |
| | | 15. x2 : Alph*
| | | 16. x3 : Alph*
| | | 17. x2 R x3
| | | ⊢  (a.x2) R (a.x3)  by (InstHyp [x2;x3;[a]] 6) THEN (Reduce -1)
| | |
| | .\
| | . 10. a : Alph
| | . 11. p : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| | . 12. x1 : x,y:Alph*//(x R y)
| | . 13. y1 : x,y:Alph*//(x R y)
| | . 14. p = <x1,y1>
| | . ⊢  a.y1 ∈ x,y:Alph*//(x R y)  by (Analyze 13) THEN Analyze
| | . |
| | . 15. y2 : Alph*
| | . 16. y3 : Alph*
| | . 17. y2 R y3
| | . ⊢  (a.y2) R (a.y3)  by (InstHyp [y2;y3;[a]] 6) THEN (Reduce -1)
| |
| .\
| . 10. <(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . |   ,λa,p. p/x,y.<a.x,a.y>>
| . |   ∈ ActionSet(Alph)
| . ⊢
| . | by Inst Thm* ∀t:T. Fin(T) ⇒ Fin(T×T)    {prod_fin_is_fin}
| . |    [x,y:Alph*//(x R y);nil]
| . |
| . |\
| . | .....wf .....
| . | ⊢  nil ∈ x,y:Alph*//(x R y)  by Analyze
| . |
| . .\
| . . 11. Fin((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y)))
| . . ⊢
| . . | by Assert
| . . |    (∀x:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))), y:Alph*.
| . . |     (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . |      ,λa,p. p/x,y.<a.x,a.y>>:y←x)
| . . |     =
| . . |     (x/x1,x2.<y@q x1,y@q x2>))
| . . |
| . . |\
| . . | .....assertion .....
| . . | ⊢  ∀x:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))), y:Alph*.
```

```
| . . | |  (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . | |   ,λa,p. p/x,y.<a.x,a.y>>:y←x)
| . . | |  =
| . . | |  (x/x1,x2.<y@q x1,y@q x2>)  by RepD
| . . | |
| . . | 12. x1 : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . | 13. y1 : Alph*
| . . | ⊢  (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . | |   ,λa,p. p/x,y.<a.x,a.y>>:y1←x1)
| . . | |  =
| . . | |  (x1/x1,x2.<y1@q x1,y1@q x2>)
| . . | | by ListInd -1 THEN Unfold ‘mn_quo_append‘ 0
| . . | |    THEN
| . . | |    RecUnfold ‘maction‘ 0
| . . | |    THEN
| . . | |    Analyze 12
| . . | |    THEN
| . . | |    Reduce 0
| . . | |
| . . | |\
| . . | | 12. x2 : x,y:Alph*//(x R y)
| . . | | 13. x3 : x,y:Alph*//(x R y)
| . . | | 14. Alph*
| . . | | ⊢  <x2,x3> = <x2,x3>  by Auto
| . . | |
| . . | .\
| . . | . 12. x2 : x,y:Alph*//(x R y)
| . . | . 13. x3 : x,y:Alph*//(x R y)
| . . | . 14. Alph*
| . . | . 15. u : Alph
| . . | . 16. v : Alph*
| . . | . 17. (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . | . |    ,λa,p. p/x,y.<a.x,a.y>>:v←<x2,x3>)
| . . | . |   =
| . . | . |   (<x2,x3>/x1,x2.<v@q x1,v@q x2>)
| . . | . ⊢  ((<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . | . |    ,λa,p. p/x,y.<a.x,a.y>>:v←<x2,x3>)
| . . | . |   /x,y.<u.x,u.y>)
| . . | . |  =
| . . | . |  <u.(v @ x2),u.(v @ x3)>
| . . | . | by (RWH (HypC -1) 0) THEN (Reduce 0)
| . . | . |    THEN
| . . | . |    (Unfold ‘mn_quo_append‘ 0)
| . . | . |
| . . | . |\
| . . | . | .....wf .....
| . . | . | ⊢  (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . | . |    =
| . . | . |    (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))  by Auto
| . . | . |
| . . | . |\
| . . | . | .....wf .....
| . . | . | 18. x4 : x,y:Alph*//(x R y)
| . . | . | 19. y2 : x,y:Alph*//(x R y)
| . . | . | 20. (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
```

```
| . . | . | |    ,λa,p. p/x,y.<a.x,a.y>>:v←<x2,x3>)
| . . | . | |   =
| . . | . | |   <x4,y2>
| . . | . | ⊢  <u.x4,u.y2> = <u.x4,u.y2>  by Analyze
| . . | . | |
| . . | . | |\
| . . | . | | ⊢  u.x4 = u.x4  by (Analyze 18) THEN Analyze
| . . | . | | |
| . . | . | | 21. x5 : Alph*
| . . | . | | 22. x6 : Alph*
| . . | . | | 23. x5 R x6
| . . | . | | ⊢  (u.x5) R (u.x6)
| . . | . | |   by (InstHyp [x5;x6;[u]] 6) THEN (Reduce -1)
| . . | . | |
| . . | . | .\
| . . | . | . ⊢  u.y2 = u.y2  by (Analyze 19) THEN Analyze
| . . | . | . |
| . . | . | . 21. y3 : Alph*
| . . | . | . 22. y4 : Alph*
| . . | . | . 23. y3 R y4
| . . | . | . ⊢  (u.y3) R (u.y4)
| . . | . | .   by (InstHyp [y3;y4;[u]] 6) THEN (Reduce -1)
| . . | . |
| . . | . |\
| . . | . | .....wf .....
| . . | . | ⊢  <u.(v @ x2),u.(v @ x3)> = <u.(v @ x2),u.(v @ x3)>
| . . | . | | by Analyze
| . . | . | |
| . . | . | |\
| . . | . | | ⊢  u.(v @ x2) = u.(v @ x2)
| . . | . | | | by (Analyze 12) THEN Analyze
| . . | . | | |
| . . | . | | 18. x4 : Alph*
| . . | . | | 19. x5 : Alph*
| . . | . | | 20. x4 R x5
| . . | . | | ⊢  (u.(v @ x4)) R (u.(v @ x5))
| . . | . | |   by (InstHyp [x4;x5;u.v] 6) THEN (Reduce -1)
| . . | . | |
| . . | . | .\
| . . | . | . ⊢  u.(v @ x3) = u.(v @ x3)
| . . | . | . | by (Analyze 13) THEN Analyze
| . . | . | . |
| . . | . | . 18. x4 : Alph*
| . . | . | . 19. x5 : Alph*
| . . | . | . 20. x4 R x5
| . . | . | . ⊢  (u.(v @ x4)) R (u.(v @ x5))
| . . | . | .   by (InstHyp [x4;x5;u.v] 6) THEN (Reduce -1)
| . . | . |
| . . | . .\
| . . | . . ⊢  <u.(v @ x2),u.(v @ x3)> = <u.(v @ x2),u.(v @ x3)>
| . . | . . | by Analyze
| . . | . . |
| . . | . . |\
| . . | . . | ⊢  u.(v @ x2) = u.(v @ x2)
| . . | . . | | by (Analyze 12) THEN Analyze
```

```
| . . | . . | |
| . . | . . | 18. x4 : Alph*
| . . | . . | 19. x5 : Alph*
| . . | . . | 20. x4 R x5
| . . | . . | ⊢  (u.(v @ x4)) R (u.(v @ x5))
| . . | . . |   by (InstHyp [x4;x5;u.v] 6) THEN (Reduce -1)
| . . | . . |
| . . | . . .\
| . . | . . . ⊢  u.(v @ x3) = u.(v @ x3)
| . . | . . . | by (Analyze 13) THEN Analyze
| . . | . . . |
| . . | . . . 18. x4 : Alph*
| . . | . . . 19. x5 : Alph*
| . . | . . . 20. x4 R x5
| . . | . . . ⊢  (u.(v @ x4)) R (u.(v @ x5))
| . . | . . .   by (InstHyp [x4;x5;u.v] 6) THEN (Reduce -1)
| . . |
| . . .\
| . . . 12. ∀x:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))), y:Alph*.
| . . . |   (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . |    ,λa,p. p/x,y.<a.x,a.y>>:y←x)
| . . . |   =
| . . . |   (x/x1,x2.<y@q x1,y@q x2>)
| . . . ⊢
| . . . | by Inst
| . . . |    Thm* ∀S:ActionSet(Alph), si:S.car.
| . . . |         Fin(S.car) ⇒
| . . . |         Fin(Alph) ⇒
| . . . |         (∃RL:S.car*.
| . . . |          ∀s:S.car.
| . . . |           (∃w:Alph*. (S:w←si) = s) ⟺ mem_f(S.car;s;RL))
| . . . |    {reach_aux}
| . . . |    [Alph
| . . . |    ;<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . |     ,λa,p. p/x,y.<a.x,a.y>>
| . . . |    ;<x,y>]
| . . . |
| . . . |\
| . . . | .....antecedent .....
| . . . | ⊢  Fin(<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . |        ,λa,p. p/x,y.<a.x,a.y>>.car)
| . . . |   by Unfold ‘aset_car‘ 0 THEN Reduce 0
| . . . |
| . . . .\
| . . . . 13. ∃RL:<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . |       ,λa,p. p/x,y.<a.x,a.y>>.car*.
| . . . . |
| . . . . |   ∀s:<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . |      ,λa,p. p/x,y.<a.x,a.y>>.car.
| . . . . |
| . . . . |   (∃w:Alph*.
| . . . . |    (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . |     ,λa,p. p/x,y.<a.x,a.y>>:w←<x,y>)
| . . . . |    =
| . . . . |    s)
```

```
| . . . . |   ⟺
| . . . . |   mem_f(<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . |         ,λa,p. p/x,y.<a.x,a.y>>.car;s;RL)
| . . . . ⊢   by Reduce -1 THEN Analyze -1
| . . . . |
| . . . . 13. RL : ((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y)))*
| . . . . 14. ∀s:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . |   (∃w:Alph*.
| . . . . |    (<(x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . |     ,λa,p. p/x,y.<a.x,a.y>>:w←<x,y>)
| . . . . |    =
| . . . . |    s)
| . . . . |   ⟺
| . . . . |   mem_f((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y));s;RL)
| . . . . ⊢   by RWH (HypC 12) -1
| . . . . |
| . . . . 14. ∀s:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . |   (∃w:Alph*. (<x,y>/x1,x2.<w@q x1,w@q x2>) = s)
| . . . . |   ⟺
| . . . . |   mem_f((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y));s;RL)
| . . . . ⊢
| . . . . | by Assert
| . . . . |    ((∃x@0:Alph*. (g(x@0@q x)) =b (g(x@0@q y)) = false2)
| . . . . |     ⟺
| . . . . |     (∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . |      (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . |      & mem_f((x,y:Alph*//(x R y))×
| . . . . |        (x,y:Alph*//(x R y));p;RL)))
| . . . . |
| . . . . |\
| . . . . | .....assertion .....
| . . . . | ⊢  (∃x@0:Alph*. (g(x@0@q x)) =b (g(x@0@q y)) = false2)
| . . . . | |  ⟺
| . . . . | |  (∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . | |   (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . | |   & mem_f((x,y:Alph*//(x R y))×
| . . . . | |     (x,y:Alph*//(x R y));p;RL))  by GenRepD
| . . . . | |
| . . . . | |\
| . . . . | | 15. ∃x@0:Alph*. (g(x@0@q x)) =b (g(x@0@q y)) = false2
| . . . . | | ⊢  ∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . | | |  (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . | | |  & mem_f((x,y:Alph*//(x R y))×
| . . . . | | |    (x,y:Alph*//(x R y));p;RL)
| . . . . | | | by Analyze -1 THEN InstConcl [<x@0@q x,x@0@q y>]
| . . . . | | |
| . . . . | | 15. x@0 : Alph*
| . . . . | | 16. (g(x@0@q x)) =b (g(x@0@q y)) = false2
| . . . . | | ⊢  (<x@0@q x,x@0@q y>/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . | | |  & mem_f((x,y:Alph*//(x R y))×
| . . . . | | |    (x,y:Alph*//(x R y));<x@0@q x,x@0@q y>;RL)
| . . . . | | | by Reduce 0
| . . . . | | |
| . . . . | | ⊢  mem_f((x,y:Alph*//(x R y))×
| . . . . | | |  (x,y:Alph*//(x R y));<x@0@q x,x@0@q y>;RL)
```

```
| . . . . | | | by BackThru 14
| . . . . | | |
| . . . . | | ⊢  ∃w:Alph*.
| . . . . | |    (<x,y>/x1,x2.<w@_q x1,w@_q x2>) = <x@0@_q x,x@0@_q y>
| . . . . | |   by (InstConcl [x@0]) THEN (Reduce 0)
| . . . . | |
| . . . . | .\
| . . . . | . 15. ∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . | . |   (p/p1,p2.(g(p1)) =_b (g(p2))) = false_2
| . . . . | . |   & mem_f((x,y:Alph*//(x R y))×
| . . . . | . |     (x,y:Alph*//(x R y));p;RL)
| . . . . | . ⊢  ∃x@0:Alph*. (g(x@0@_q x)) =_b (g(x@0@_q y)) = false_2
| . . . . | . | by Analyze -1 THEN Analyze -1
| . . . . | . |
| . . . . | . 15. p : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . | . 16. (p/p1,p2.(g(p1)) =_b (g(p2))) = false_2
| . . . . | . 17. mem_f((x,y:Alph*//(x R y))×
| . . . . | . |   (x,y:Alph*//(x R y));p;RL)
| . . . . | . ⊢   by RWH (RevHypC 14) -1
| . . . . | . |
| . . . . | . 17. ∃w:Alph*. (<x,y>/x1,x2.<w@_q x1,w@_q x2>) = p
| . . . . | . ⊢   by Analyze -1 THEN InstConcl [w]
| . . . . | . |
| . . . . | . 17. w : Alph*
| . . . . | . 18. (<x,y>/x1,x2.<w@_q x1,w@_q x2>) = p
| . . . . | . ⊢  (g(w@_q x)) =_b (g(w@_q y)) = false_2
| . . . . | .   by (Reduce -1 THEN RWH (RevHypC -1) 16)
| . . . . | .      THEN
| . . . . | .      (Reduce -1)
| . . . . |
| . . . . .\
| . . . . . 15. (∃x@0:Alph*. (g(x@0@_q x)) =_b (g(x@0@_q y)) = false_2)
| . . . . . |   ⟺
| . . . . . |   (∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . |    (p/p1,p2.(g(p1)) =_b (g(p2))) = false_2
| . . . . . |    & mem_f((x,y:Alph*//(x R y))×
| . . . . . |      (x,y:Alph*//(x R y));p;RL))
| . . . . . ⊢
| . . . . . | by (RWH (HypC -1) 0) THEN (Thin 15) THEN (Thin 14)
| . . . . . |    THEN
| . . . . . |    (Thin 12)
| . . . . . |    THEN
| . . . . . |    (Thin 10)
| . . . . . |
| . . . . . 10. Fin((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y)))
| . . . . . 11. RL : ((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y)))*
| . . . . . ⊢  Dec(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . |  (p/p1,p2.(g(p1)) =_b (g(p2))) = false_2
| . . . . . |  & mem_f((x,y:Alph*//(x R y))×
| . . . . . |    (x,y:Alph*//(x R y));p;RL))  by ListInd 11
| . . . . . |
| . . . . . |\
| . . . . . | ⊢  Dec(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . | |  (p/p1,p2.(g(p1)) =_b (g(p2))) = false_2
| . . . . . | |  & mem_f((x,y:Alph*//(x R y))×
```

```
| . . . . . | |    (x,y:Alph*//(x R y));p;nil))  by Reduce 0
| . . . . . | |
| . . . . . | ⊢  Dec(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . |    (p/p1,p2.(g(p1)) =b (g(p2))) = false2 & False)
| . . . . . |   by (Sel 2 (Analyze 0)) THEN (Analyze 0)
| . . . . . |      THEN
| . . . . . |      (Analyze -1)
| . . . . . |      THEN
| . . . . . |      ProveProp
| . . . . . |
| . . . . . .\
| . . . . . . 12. u : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . . . 13. v : ((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y)))*
| . . . . . . 14. Dec(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . . |   (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . |   & mem_f((x,y:Alph*//(x R y))×
| . . . . . . |     (x,y:Alph*//(x R y));p;v))
| . . . . . . ⊢  Dec(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . . |  (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . |  & mem_f((x,y:Alph*//(x R y))×
| . . . . . . |    (x,y:Alph*//(x R y));p;u.v))
| . . . . . . | by Reduce 0
| . . . . . . |    THEN
| . . . . . . |    Decide ((u/u1,u2.(g(u1)) =b (g(u2))) = false2)
| . . . . . . |
| . . . . . . |\
| . . . . . . | 15. (u/u1,u2.(g(u1)) =b (g(u2))) = false2
| . . . . . . | ⊢  Dec(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . . | |
| . . . . . . | |  (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . | |  & u = p
| . . . . . . | |    ∨ mem_f((x,y:Alph*//(x R y))×
| . . . . . . | |      (x,y:Alph*//(x R y));p;v))
| . . . . . . | | by Sel 1 (Analyze 0)
| . . . . . . | |
| . . . . . . | ⊢  ∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . . | |  (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . | |  & u = p
| . . . . . . | |    ∨ mem_f((x,y:Alph*//(x R y))×
| . . . . . . | |      (x,y:Alph*//(x R y));p;v)  by InstConcl [u]
| . . . . . . | |
| . . . . . . | ⊢  u = u
| . . . . . . |    ∨ mem_f((x,y:Alph*//(x R y))×
| . . . . . . |      (x,y:Alph*//(x R y));u;v)
| . . . . . . |   by (Sel 1 (Analyze 0)) THEN Eq
| . . . . . . |
| . . . . . . .\
| . . . . . . . 15. ¬(u/u1,u2.(g(u1)) =b (g(u2))) = false2
| . . . . . . . ⊢  Dec(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . . . |
| . . . . . . . |  (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . . |  & u = p
| . . . . . . . |    ∨ mem_f((x,y:Alph*//(x R y))×
| . . . . . . . |      (x,y:Alph*//(x R y));p;v))  by Analyze -2
| . . . . . . . |
```

```
| . . . . . |\
| . . . . . | 14. ∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . | |   (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . | |   & mem_f((x,y:Alph*//(x R y))×
| . . . . . | |     (x,y:Alph*//(x R y));p;v)
| . . . . . | 15. ¬(u/u1,u2.(g(u1)) =b (g(u2))) = false2
| . . . . . | ⊢   by Analyze 14 THEN Sel 1 (Analyze 0)
| . . . . . | |
| . . . . . | 14. p : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . . | 15. (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . | |   & mem_f((x,y:Alph*//(x R y))×
| . . . . . | |     (x,y:Alph*//(x R y));p;v)
| . . . . . | 16. ¬(u/u1,u2.(g(u1)) =b (g(u2))) = false2
| . . . . . | ⊢  ∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . | |  (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . | |  & u = p
| . . . . . | |    ∨ mem_f((x,y:Alph*//(x R y))×
| . . . . . | |      (x,y:Alph*//(x R y));p;v)  by InstConcl [p]
| . . . . . | |
| . . . . . | 15. (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . | 16. mem_f((x,y:Alph*//(x R y))×
| . . . . . |     (x,y:Alph*//(x R y));p;v)
| . . . . . | 17. ¬(u/u1,u2.(g(u1)) =b (g(u2))) = false2
| . . . . . | ⊢  u = p
| . . . . . |    ∨ mem_f((x,y:Alph*//(x R y))×
| . . . . . |      (x,y:Alph*//(x R y));p;v)
| . . . . . |   by (Sel 2 (Analyze 0)) THEN (NthHyp -2)
| . . . . . |
| . . . . . .\
| . . . . . . 14. ¬(∃p:((x,y:Alph*//(x R y))×
| . . . . . . |         (x,y:Alph*//(x R y))).
| . . . . . . |
| . . . . . . |     (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . |     & mem_f((x,y:Alph*//(x R y))×
| . . . . . . |       (x,y:Alph*//(x R y));p;v))
| . . . . . . 15. ¬(u/u1,u2.(g(u1)) =b (g(u2))) = false2
| . . . . . . ⊢   by Sel 2 (Analyze 0)
| . . . . . . |
| . . . . . . ⊢  ¬(∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . . . |
| . . . . . . |    (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . |    & u = p
| . . . . . . |      ∨ mem_f((x,y:Alph*//(x R y))×
| . . . . . . |        (x,y:Alph*//(x R y));p;v))
| . . . . . . | by (Analyze 0) THEN (RepeatFor 3 (Analyze -1))
| . . . . . . |
| . . . . . . |\
| . . . . . . | 16. p : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . . . | 17. (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . . . | 18. u = p
| . . . . . . | ⊢  False  by (RWH (HypC -1) 15) THEN (Analyze 15)
| . . . . . . |
| . . . . . . .\
| . . . . . . . 16. p : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . . . . 17. (p/p1,p2.(g(p1)) =b (g(p2))) = false2
```

```
| . . . . . 18. mem_f((x,y:Alph*//(x R y))×
| . . . . . |   (x,y:Alph*//(x R y));p;v)
| . . . . . ⊢  False  by Analyze 14
| . . . . . |
| . . . . . 14. ¬(u/u1,u2.(g(u1)) =b (g(u2))) = false2
| . . . . . 15. p : (x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))
| . . . . . 16. (p/p1,p2.(g(p1)) =b (g(p2))) = false2

| . . . . . 17. mem_f((x,y:Alph*//(x R y))×
| . . . . .     (x,y:Alph*//(x R y));p;v)
| . . . . . ⊢  ∃p:((x,y:Alph*//(x R y))×(x,y:Alph*//(x R y))).
| . . . . .
| . . . . .    (p/p1,p2.(g(p1)) =b (g(p2))) = false2
| . . . . .    & mem_f((x,y:Alph*//(x R y))×
| . . . . .      (x,y:Alph*//(x R y));p;v)  by InstConcl [p]
|
.\
. 10. Dec(∀x@0:Alph*. (λz.g(z@q x) = g(z@q y))(x@0))
. ⊢   by Reduce -1
END
```