```
iseg_select  THM

⊢  ∀T:Type, l1,l2:T List.
|  l1 ≤ l2 ⟺ ||l1||≤||l2|| & (∀i:ℕ. i<||l1|| ⇒ l1[i] = l2[i])
| by D 0 THENM D 0 THENA Auto THEN ListInd (-1)
|
|\
| 1. T : Type
| 2. T List
| ⊢  ∀l2:T List.
|     nil ≤ l2 ⟺ ||nil||≤||l2|| & (∀i:ℕ. i<||nil|| ⇒ nil[i] = l2[i])
|    by Reduce 0 THEN Auto' THEN Try Easy
|
.\
. 1. T : Type
. 2. T List
. 3. u : T
. 4. v : T List
. 5. ∀l2:T List.
. |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. ⊢  ∀l2:T List.
. |  [u / v] ≤ l2
. |  ⟺
. |  ||[u / v]||≤||l2|| & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = l2[i])
. | by D 0 THENA Auto THEN ListInd (-1)
. |
. |\
. | 1. T : Type
. | 2. T List
. | 3. u : T
. | 4. v : T List
. | 5. ∀l2:T List.
. | |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. | 6. T List
. | ⊢  [u / v] ≤ nil
. | |  ⟺
. | |  ||[u / v]||≤||nil||
. | |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = nil[i])
. | | by Reduce 0 THEN D 0 THEN D 0 THENA Auto
. | |
. | |
. | |\
. | | 1. T : Type
. | | 2. T List
. | | 3. u : T
. | | 4. v : T List
. | | 5. ∀l2:T List.
. | | |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. | | 6. T List
. | | 7. [u / v] ≤ nil
. | | ⊢  ||v||+1≤0 & (∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = nil[i])
. | | | by Unfold `iseg` (-1) THEN ExRepD
. | | |
. | | 1. T : Type
. | | 2. T List
```

```
. | | 3. u : T
. | | 4. v : T List
. | | 5. ∀l2:T List.
. | |    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. | | 6. T List
. | | 7. l : T List
. | | 8. nil = ([u / v] @ l)
. | | ⊢  ||v||+1≤0 & (∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = nil[i])
. | |   by ApFunToHypEquands ‘z’ ||z|| ℕ
. | |       (-1) THEN Reduce (-1) THEN Auto’
. | |
. | .\
. | . 1. T : Type
. | . 2. T List
. | . 3. u : T
. | . 4. v : T List
. | . 5. ∀l2:T List.
. | .    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. | . 6. T List
. | . 7. ||v||+1≤0 & (∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = nil[i])
. | . ⊢  [u / v] ≤ nil  by D (-1) THEN Auto’
. |
. .\
. . 1. T : Type
. . 2. T List
. . 3. u : T
. . 4. v : T List
. . 5. ∀l2:T List.
. . |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . 6. T List
. . 7. u1 : T
. . 8. v1 : T List
. . 9. [u / v] ≤ v1
. . |  ⟺
. . |  ||[u / v]||≤||v1||
. . |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . ⊢  [u / v] ≤ [u1 / v1]
. . |  ⟺
. . |  ||[u / v]||≤||[u1 / v1]||
. . |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = [u1 / v1][i])
. . | by Reduce 0 THEN D 0 THENM D 0 THENA Auto
. . |
. . |\
. . | 1. T : Type
. . | 2. T List
. . | 3. u : T
. . | 4. v : T List
. . | 5. ∀l2:T List.
. . | |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | 6. T List
. . | 7. u1 : T
. . | 8. v1 : T List
. . | 9. [u / v] ≤ v1
. . | |  ⟺
. . | |  ||[u / v]||≤||v1||
```

```
. . | |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | 10. [u / v] ≤ [u1 / v1]
. . | ⊢  ||v||+1≤||v1||+1
. . | |  & (∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = [u1 / v1][i])
. . | | by RWW "cons_iseg" (-1) THENA Auto THEN D (-1)
. . |
. . | |       THEN AllHyps (InstHyp [v1]) THENA Auto
. . | |
. . | 1. T : Type
. . | 2. T List
. . | 3. u : T
. . | 4. v : T List
. . | 5. ∀l2:T List.
. . | |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | 6. T List
. . | 7. u1 : T
. . | 8. v1 : T List
. . | 9. [u / v] ≤ v1
. . | |  ⟺
. . | |  ||[u / v]||≤||v1||
. . | |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | 10. u = u1
. . | 11. v ≤ v1
. . | 12. [u / v] ≤ v1
. . | |   ⟺
. . | |   ||[u / v]||≤||v1||
. . | |   & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | 13. v ≤ v1 ⟺ ||v||≤||v1|| & (∀i:ℕ. i<||v|| ⇒ v[i] = v1[i])
. . | ⊢  ||v||+1≤||v1||+1
. . | |  & (∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = [u1 / v1][i])
. . | | by ...
. . | |
. . | 1. T : Type
. . | 2. T List
. . | 3. u : T
. . | 4. v : T List
. . | 5. ∀l2:T List.
. . | |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | 6. T List
. . | 7. u1 : T
. . | 8. v1 : T List
. . | 9. [u / v] ≤ v1
. . | |  ⇒
. . | |  ||[u / v]||≤||v1||
. . | |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | |   & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | 11. u = u1
. . | 12. v ≤ v1
. . | 13. [u / v] ≤ v1
. . | |   ⇒
. . | |   ||[u / v]||≤||v1||
. . | |   & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | 14. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | |   & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
```

```
. . | 15. ||v||≤||v1||
. . | 16. ∀i:ℕ. i<||v|| ⇒ v[i] = v1[i]
. . | 17. i : ℕ
. . | 18. i<||v||+1
. . | ⊢  [u / v][i] = [u1 / v1][i]  by CaseNat 0 ‘i’
. . | |
. . | |\
. . | | 1. T : Type
. . | | 2. T List
. . | | 3. u : T
. . | | 4. v : T List
. . | | 5. ∀l2:T List.
. . | |    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | | 6. T List
. . | | 7. u1 : T
. . | | 8. v1 : T List
. . | | 9. [u / v] ≤ v1
. . | |    ⇒
. . | |    ||[u / v]||≤||v1||
. . | |    & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | | 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | |     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | | 11. u = u1
. . | | 12. v ≤ v1
. . | | 13. [u / v] ≤ v1
. . | |     ⇒
. . | |     ||[u / v]||≤||v1||
. . | |     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | | 14. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | |     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | | 15. ||v||≤||v1||
. . | | 16. ∀i:ℕ. i<||v|| ⇒ v[i] = v1[i]
. . | | 17. i : ℕ
. . | | 18. i<||v||+1
. . | | 19. i = 0
. . | | ⊢  [u / v][0] = [u1 / v1][0]
. . | |   by RWW "select_cons_hd" 0 THEN Auto

. . | |
. . | .\
. . | . 1. T : Type
. . | . 2. T List
. . | . 3. u : T
. . | . 4. v : T List
. . | . 5. ∀l2:T List.
. . | .    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | . 6. T List
. . | . 7. u1 : T
. . | . 8. v1 : T List
. . | . 9. [u / v] ≤ v1
. . | .    ⇒
. . | .    ||[u / v]||≤||v1||
. . | .    & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | . 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | .     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | . 11. u = u1
```

```
. . | . 12. v ≤ v1
. . | . 13. [u / v] ≤ v1
. . | .     ⇒
. . | .     ||[u / v]||≤||v1||
. . | .     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | . 14. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | .     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | . 15. ||v||≤||v1||
. . | . 16. ∀i:ℕ. i<||v|| ⇒ v[i] = v1[i]
. . | . 17. i : ℕ
. . | . 18. i<||v||+1
. . | . 19. ¬i = 0
. . | . ⊢  [u / v][i] = [u1 / v1][i]
. . | .   by RWW "select_cons_tl" 0 THEN Auto THEN Easy
. . |
. . |\
. . | 1. T : Type
. . | 2. T List
. . | 3. u : T
. . | 4. v : T List
. . | 5. ∀l2:T List.
. . | |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | 6. T List
. . | 7. u1 : T
. . | 8. v1 : T List
. . | 9. [u / v] ≤ v1
. . | |  ⟺
. . | |  ||[u / v]||≤||v1||
. . | |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | 10. ||v||+1≤||v1||+1
. . | |   & (∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = [u1 / v1][i])
. . | ⊢  [u / v] ≤ [u1 / v1]
. . | | by D (-1) THEN RWW "cons_iseg" 0 THEN Auto
. . | |
. . | |\
. . | | 1. T : Type
. . | | 2. T List
. . | | 3. u : T
. . | | 4. v : T List
. . | | 5. ∀l2:T List.
. . | |    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | | 6. T List
. . | | 7. u1 : T
. . | | 8. v1 : T List
. . | | 9. [u / v] ≤ v1
. . | |    ⇒
. . | |    ||[u / v]||≤||v1||
. . | |    & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | | 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | |     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | | 11. ||v||+1≤||v1||+1
. . | | 12. ∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = [u1 / v1][i]
. . | | ⊢  u = u1  by InstHyp [0] (-1) THEN Reduce (-1) THEN Auto'
. . | |
. . | .\
```

```
. . | . 1. T : Type
. . | . 2. T List
. . | . 3. u : T
. . | . 4. v : T List
. . | . 5. ∀l2:T List.
. . | . |  v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | . 6. T List
. . | . 7. u1 : T
. . | . 8. v1 : T List
. . | . 9. [u / v] ≤ v1
. . | . |  ⇒
. . | . |  ||[u / v]||≤||v1||
. . | . |  & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | . 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | . |   & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | . 11. ||v||+1≤||v1||+1
. . | . 12. ∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = [u1 / v1][i]
. . | . ⊢  v ≤ v1  by BackThruSomeHyp THEN Auto
. . | . |
. . | . |
. . | . 1. T : Type
. . | . 2. T List
. . | . 3. u : T
. . | . 4. v : T List
. . | . 5. ∀l2:T List.
. . | .    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | . 6. T List
. . | . 7. u1 : T
. . | . 8. v1 : T List
. . | . 9. [u / v] ≤ v1
. . | .    ⇒
. . | .    ||[u / v]||≤||v1||
. . | .    & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | . 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . | .     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | . 11. ||v||+1≤||v1||+1
. . | . 12. ∀i:ℕ. i<||v||+1 ⇒ [u / v][i] = [u1 / v1][i]
. . | . 13. i : ℕ
. . | . 14. i<||v||
. . | . ⊢  v[i] = v1[i]
. . | .   by InstHyp [i+1] (-3)  THENM RWW "select_cons_tl" (-1)
. . | .
. . | .        THENM ArithSimp (-1) THEN Auto
. . |
. . |\
. . | 1. T : Type
. . | 2. T List
. . | 3. u : T
. . | 4. v : T List
. . | 5. ∀l2:T List.
. . |    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . | 6. T List
. . | 7. T
. . | 8. v1 : T List
. . | 9. [u / v] ≤ v1
```

```
. . |    ⇒
. . |    ||[u / v]||≤||v1||
. . |    & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . | 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . |     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . | 11. ||v||+1≤||v1||+1
. . | 12. i : ℕ
. . | 13. i<||v||+1
. . | ⊢  i<||[u / v]||  by Reduce 0 THEN Auto
. . |
. . .\
. . . 1. T : Type
. . . 2. T List
. . . 3. u : T
. . . 4. v : T List
. . . 5. ∀l2:T List.
. . .    v ≤ l2 ⟺ ||v||≤||l2|| & (∀i:ℕ. i<||v|| ⇒ v[i] = l2[i])
. . . 6. T List
. . . 7. u1 : T
. . . 8. v1 : T List
. . . 9. [u / v] ≤ v1
. . .    ⇒
. . .    ||[u / v]||≤||v1||
. . .    & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i])
. . . 10. [u / v] ≤ v1 ⇐ (||[u / v]||≤||v1||
. . .     & (∀i:ℕ. i<||[u / v]|| ⇒ [u / v][i] = v1[i]))
. . . 11. ||v||+1≤||v1||+1
. . . 12. i : ℕ
. . . 13. i<||v||+1
. . . ⊢  i<||[u1 / v1]||  by Reduce 0 THEN Auto
END
```