```
count_pairs_swap  THM

⊢  ∀T:Type, L:T List, P:(T→T→𝔹), n:ℕ(||L||-1).
|  count(x < y in swap(L;n;n+1) | P(x,y))
|  =
|  count(x < y in L | P(x,y))+if P(L[n],L[(n+1)])→ -1 else 0 fi
|  +if P(L[(n+1)],L[n])→ 1 else 0 fi
| by Auto THEN Unfold `count_pairs` 0
THEN Subst' ||swap(L;n;n+1)||
|                                                     =
|                                                     ||L|| 0

|    THENA Auto
THEN Try (BackThruLemma `swap_length` THEN Auto)
|
1. T : Type
2. L : T List
3. P : T→T→𝔹
4. n : ℕ(||L||-1)
⊢  sum(if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1
|      else 0 fi | i < ||L||; j < ||L||)
|  =
|  sum(if (i<₂j)∧₂P(L[i],L[j])→ 1 else 0 fi | i < ||L||; j < ||L||)
|  +if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1 else 0 fi
| by AssertBY (n, n+1) ∈ ℕ||L||→ℕ||L|| Auto'
|
1. T : Type
2. L : T List
3. P : T→T→𝔹
4. n : ℕ(||L||-1)
5. (n, n+1) ∈ ℕ||L||→ℕ||L||
⊢  sum(if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1
|      else 0 fi | i < ||L||; j < ||L||)
|  =
|  sum(if (i<₂j)∧₂P(L[i],L[j])→ 1 else 0 fi | i < ||L||; j < ||L||)
|  +if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1 else 0 fi
| by AssertBY ||swap(L;n;n+1)|| = ||L||

|    (BackThruLemma `swap_length` THEN Auto)
|
1. T : Type
2. L : T List
3. P : T→T→𝔹
4. n : ℕ(||L||-1)
5. (n, n+1) ∈ ℕ||L||→ℕ||L||
6. ||swap(L;n;n+1)|| = ||L||
⊢  sum(if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1
|      else 0 fi | i < ||L||; j < ||L||)
|  =
|  sum(if (i<₂j)∧₂P(L[i],L[j])→ 1 else 0 fi | i < ||L||; j < ||L||)
|  +if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1 else 0 fi
| by AssertBY ∃f:(ℕ||L||→ℕ||L||). ∀i:ℕ||L||. f(i) = (n, n+1)(i)

|     (InstConcl [(n, n+1)] THEN Auto)
THEN ExRepD
```

```
|
1. T : Type
2. L : T List
3. P : T→T→𝔹
4. n : ℕ(||L||-1)
5. (n, n+1) ∈ ℕ||L||→ℕ||L||
6. ||swap(L;n;n+1)|| = ||L||
7. f : ℕ||L||→ℕ||L||
8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
⊢  sum(if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1
|      else 0 fi | i < ||L||; j < ||L||)
|  =
|  sum(if (i<₂j)∧₂P(L[i],L[j])→ 1 else 0 fi | i < ||L||; j < ||L||)
|  +if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1 else 0 fi
| by Subst' sum(if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1
|               else 0 fi | i < ||L||; j < ||L||)
|           =
|           sum(if (f(i)<₂f(j))∧₂P(swap(L;n;n+1)[(f(i))]
|                                 ,swap(L;n;n+1)[(f(j))])→
|                  1
|               else 0 fi | i < ||L||; j < ||L||)
|     0 THEN Try (Complete Auto')
|
|\
| .....equality .....
| 1. T : Type
| 2. L : T List
| 3. P : T→T→𝔹
| 4. n : ℕ(||L||-1)
| 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
| 6. ||swap(L;n;n+1)|| = ||L||
| 7. f : ℕ||L||→ℕ||L||
| 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
| ⊢  sum(if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1
| |      else 0 fi | i < ||L||; j < ||L||)
| |  =
| |  sum(if (f(i)<₂f(j))∧₂P(swap(L;n;n+1)[(f(i))]
| |                         ,swap(L;n;n+1)[(f(j))])→
| |         1
| |      else 0 fi | i < ||L||; j < ||L||)
| | by Assert Bij(ℕ||L||; ℕ||L||; f)
| |
| |\
| | .....assertion .....
| | 1. T : Type
| | 2. L : T List
| | 3. T→T→𝔹
| | 4. n : ℕ(||L||-1)
| | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
| | 6. ||swap(L;n;n+1)|| = ||L||
| | 7. f : ℕ||L||→ℕ||L||
| | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
| | ⊢  Bij(ℕ||L||; ℕ||L||; f)  by Subst f = (n, n+1) 0 THEN Auto
| | |
| | |\
```

```
| | | .....equality .....
| | | 1. T : Type
| | | 2. L : T List
| | | 3. T→T→𝔹
| | | 4. n : ℕ(||L||-1)
| | | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
| | | 6. ||swap(L;n;n+1)|| = ||L||
| | | 7. f : ℕ||L||→ℕ||L||
| | | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
| | | ⊢  f = (n, n+1)  by Ext THEN Auto THEN EasyHyp
| | |
| | .\
| | . 1. T : Type
| | . 2. L : T List
| | . 3. T→T→𝔹
| | . 4. n : ℕ(||L||-1)
| | . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
| | . 6. ||swap(L;n;n+1)|| = ||L||
| | . 7. f : ℕ||L||→ℕ||L||
| | . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
| | . ⊢  Bij(ℕ||L||; ℕ||L||; (n, n+1))
| | .   by BackThruLemma ‘flip_bijection‘ THEN Auto’
| |
| .\
| . 1. T : Type
| . 2. L : T List
| . 3. P : T→T→𝔹
| . 4. n : ℕ(||L||-1)
| . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
| . 6. ||swap(L;n;n+1)|| = ||L||
| . 7. f : ℕ||L||→ℕ||L||
| . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
| . 9. Bij(ℕ||L||; ℕ||L||; f)
| . ⊢  sum(if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1
| .        else 0 fi | i < ||L||; j < ||L||)
| .    =
| .    sum(if (f(i)<₂f(j))∧₂P(swap(L;n;n+1)[(f(i))]
| .                           ,swap(L;n;n+1)[(f(j))])→
| .           1
| .        else 0 fi | i < ||L||; j < ||L||)
| .   by InstLemma ‘permute_double_sum‘ [||L||;||L||;λi,j.
| .      if (i<₂j)∧₂P(swap(L;n;n+1)[i],swap(L;n;n+1)[j])→ 1 else 0 fi;
| .      f;f]
| . THENM All ReduceSOAps
| . THEN Auto’
|
.\
. 1. T : Type
. 2. L : T List
. 3. P : T→T→𝔹
. 4. n : ℕ(||L||-1)
. 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. 6. ||swap(L;n;n+1)|| = ||L||
. 7. f : ℕ||L||→ℕ||L||
. 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
```

```
. ⊢  sum(if (f(i)<_2f(j))∧_2P(swap(L;n;n+1)[(f(i))]
. |                        ,swap(L;n;n+1)[(f(j))])→
. |        1
. |      else 0 fi | i < ||L||; j < ||L||)
. |  =
. |  sum(if (i<_2j)∧_2P(L[i],L[j])→ 1 else 0 fi | i < ||L||; j < ||L||)
. |  +if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1
. |                                    else 0 fi
. | by BackThruLemma ‘double_sum_difference‘ THEN Auto’
. |
. 1. T : Type
. 2. L : T List
. 3. P : T→T→𝔹
. 4. n : ℕ(||L||-1)
. 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. 6. ||swap(L;n;n+1)|| = ||L||
. 7. f : ℕ||L||→ℕ||L||
. 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. ⊢  sum(if (f(i)<_2f(j))∧_2P(swap(L;n;n+1)[(f(i))]
. |                        ,swap(L;n;n+1)[(f(j))])→
. |        1
. |      else 0 fi
. |  -if (i<_2j)∧_2P(L[i],L[j])→ 1 else 0 fi | i < ||L||; j < ||L||)
. |  =
. |  if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1
. |                                   else 0 fi
. | by Subst’ if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1
. |                                             else 0 fi
. |           =
. |           sum(if ((i=_2n)∧_2(j=_2n+1))∧_2P(L[i],L[j])→ -1
. |                ; ((i=_2n+1)∧_2(j=_2n))∧_2P(L[i],L[j])→ 1
. |               else 0 fi | i < ||L||; j < ||L||) 0
. |
. |\
. | 1. T : Type
. | 2. L : T List
. | 3. P : T→T→𝔹
. | 4. n : ℕ(||L||-1)
. | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. | 6. ||swap(L;n;n+1)|| = ||L||
. | 7. f : ℕ||L||→ℕ||L||
. | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. | ⊢  if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1
. | |                                    else 0 fi
. | |  =
. | |  sum(if ((i=_2n)∧_2(j=_2n+1))∧_2P(L[i],L[j])→ -1
. | |       ; ((i=_2n+1)∧_2(j=_2n))∧_2P(L[i],L[j])→ 1
. | |      else 0 fi | i < ||L||; j < ||L||)
. | | by InstLemma ‘pair_support_double_sum‘ [||L||;||L||;λi,j.
. | |    if ((i=_2n)∧_2(j=_2n+1))∧_2P(L[i],L[j])→ -1
. | |     ; ((i=_2n+1)∧_2(j=_2n))∧_2P(L[i],L[j])→ 1
. | |    else 0 fi;n;n+1;n+1;n] THENA Auto THEN All ReduceSOAps
. | |
. | |\
. | | 1. T : Type
```

```
. | | 2. L : T List
. | | 3. P : T→T→𝔹
. | | 4. n : ℕ(||L||-1)
. | | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. | | 6. ||swap(L;n;n+1)|| = ||L||
. | | 7. f : ℕ||L||→ℕ||L||
. | | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. | | 9. x : ℕ||L||
. | | 10. y : ℕ||L||
. | | 11. ¬(x = n & y = n+1)
. | | 12. ¬(x = n+1 & y = n)
. | | ⊢  if ((x=_2n)∧_2(y=_2n+1))∧_2(P(L[x],L[y]))→ -1
. | |     ; ((x=_2n+1)∧_2(y=_2n))∧_2(P(L[x],L[y]))→ 1
. | |    else 0 fi
. | |    =
. | |    0
. | |   by AutoBoolCase x=_2n THEN AutoBoolCase y=_2n+1 THEN
. | |
. | |     (AutoBoolCase x=_2n+1 THEN AutoBoolCase y=_2n) THEN
. | |
. | |     AllHyps (\h . D h THEN Complete Auto)
. | |
. | .\
. | . 1. T : Type
. | . 2. L : T List
. | . 3. P : T→T→𝔹
. | . 4. n : ℕ(||L||-1)
. | . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. | . 6. ||swap(L;n;n+1)|| = ||L||
. | . 7. f : ℕ||L||→ℕ||L||
. | . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. | . 9. sum(if ((x=_2n)∧_2(y=_2n+1))∧_2(P(L[x],L[y]))→ -1
. | . |      ; ((x=_2n+1)∧_2(y=_2n))∧_2(P(L[x],L[y]))→ 1
. | . |      else 0 fi | x < ||L||; y < ||L||)
. | . |  =
. | . |  if ((n=_2n)∧_2(n+1=_2n+1))∧_2(P(L[n],L[(n+1)]))→ -1
. | . |   ; ((n=_2n+1)∧_2(n+1=_2n))∧_2(P(L[n],L[(n+1)]))→ 1
. | . |  else 0 fi+if ((n+1=_2n)∧_2(n=_2n+1))∧_2(P(L[(n+1)],L[n]))→ -1
. | . |             ; ((n+1=_2n+1)∧_2(n=_2n))∧_2(P(L[(n+1)],L[n]))→ 1
. | . |            else 0 fi
. | . ⊢  if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1
. | . |                                     else 0 fi
. | . |  =
. | . |  sum(if ((i=_2n)∧_2(j=_2n+1))∧_2(P(L[i],L[j]))→ -1
. | . |      ; ((i=_2n+1)∧_2(j=_2n))∧_2(P(L[i],L[j]))→ 1
. | . |     else 0 fi | i < ||L||; j < ||L||)
. | . | by HypSubstSq (-1) 0 THEN Thin (-1)
. | . |
. | . 1. T : Type
. | . 2. L : T List
. | . 3. P : T→T→𝔹
. | . 4. n : ℕ(||L||-1)
. | . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. | . 6. ||swap(L;n;n+1)|| = ||L||
. | . 7. f : ℕ||L||→ℕ||L||
```

```
. | . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. | . ⊢  if P(L[n],L[(n+1)])→ -1 else 0 fi+if P(L[(n+1)],L[n])→ 1
. | .                                     else 0 fi
. | .    =
. | .    if ((n=_2n)∧_2(n+1=_2n+1))∧_2(P(L[n],L[(n+1)]))→ -1
. | .     ; ((n=_2n+1)∧_2(n+1=_2n))∧_2(P(L[n],L[(n+1)]))→ 1
. | .    else 0 fi+if ((n+1=_2n)∧_2(n=_2n+1))∧_2(P(L[(n+1)],L[n]))→ -1
. | .               ; ((n+1=_2n+1)∧_2(n=_2n))∧_2(P(L[(n+1)],L[n]))→ 1
. | .              else 0 fi
. | .   by AutoBoolCase n=_2n THEN AutoBoolCase n=_2n+1 THEN
. | .
. | .      AutoBoolCase n+1=_2n THEN AutoBoolCase n+1=_2n+1
. |
. .\
. . 1. T : Type
. . 2. L : T List
. . 3. P : T→T→𝔹
. . 4. n : ℕ(||L||-1)
. . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . 6. ||swap(L;n;n+1)|| = ||L||
. . 7. f : ℕ||L||→ℕ||L||
. . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . ⊢  sum(if (f(i)<_2f(j))∧_2P(swap(L;n;n+1)[(f(i))]
. . |                         ,swap(L;n;n+1)[(f(j))])→
. . |         1
. . |      else 0 fi
. . |  -if (i<_2j)∧_2P(L[i],L[j])→ 1 else 0 fi | i < ||L||; j < ||L||)
. . |  =
. . |  sum(if ((i=_2n)∧_2(j=_2n+1))∧_2P(L[i],L[j])→ -1
. . |       ; ((i=_2n+1)∧_2(j=_2n))∧_2P(L[i],L[j])→ 1
. . |      else 0 fi | i < ||L||; j < ||L||)
. . | by BackThruLemma ‘double_sum_functionality‘ THEN Auto
. . |
. . 1. T : Type
. . 2. L : T List
. . 3. P : T→T→𝔹
. . 4. n : ℕ(||L||-1)
. . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . 6. ||swap(L;n;n+1)|| = ||L||
. . 7. f : ℕ||L||→ℕ||L||
. . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . 9. i : ℕ||L||
. . 10. j : ℕ||L||
. . ⊢  if (f(i)<_2f(j))∧_2P(swap(L;n;n+1)[(f(i))]
. . |                     ,swap(L;n;n+1)[(f(j))])→
. . |     1
. . |  else 0 fi-if (i<_2j)∧_2P(L[i],L[j])→ 1 else 0 fi
. . |  =
. . |  if ((i=_2n)∧_2(j=_2n+1))∧_2P(L[i],L[j])→ -1
. . |   ; ((i=_2n+1)∧_2(j=_2n))∧_2P(L[i],L[j])→ 1
. . |  else 0 fi
. . | by Subst swap(L;n;n+1)[(f(i))] = L[i] 0 THEN Auto’
. . |
. . |
. . |
```

```
. . |\
. . | .....equality .....
. . | 1. T : Type
. . | 2. L : T List
. . | 3. T→T→𝔹
. . | 4. n : ℕ(||L||-1)
. . | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . | 6. ||swap(L;n;n+1)|| = ||L||
. . | 7. f : ℕ||L||→ℕ||L||
. . | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . | 9. i : ℕ||L||
. . | 10. ℕ||L||
. . | ⊢  swap(L;n;n+1)[(f(i))] = L[i]
. . | | by RWO "swap_select" 0 THENA Auto THEN
. . | |  Subst' f(i)
. . | |                                                  =
. . | |                                                  (n, n+1)(i)
. . | |     0 THEN EasyHyp
. . | |
. . | 1. T : Type
. . | 2. L : T List
. . | 3. T→T→𝔹
. . | 4. n : ℕ(||L||-1)
. . | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . | 6. ||swap(L;n;n+1)|| = ||L||
. . | 7. f : ℕ||L||→ℕ||L||
. . | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . | 9. i : ℕ||L||
. . | 10. ℕ||L||
. . | ⊢  L[((n, n+1)((n, n+1)(i)))] = L[i]
. . | | by Subst' (n, n+1)((n, n+1)(i)) = i 0
. . | |
. . | |\
. . | | 1. T : Type
. . | | 2. L : T List
. . | | 3. T→T→𝔹
. . | | 4. n : ℕ(||L||-1)
. . | | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . | | 6. ||swap(L;n;n+1)|| = ||L||
. . | | 7. f : ℕ||L||→ℕ||L||
. . | | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . | | 9. i : ℕ||L||
. . | | 10. ℕ||L||
. . | | ⊢  (n, n+1)((n, n+1)(i)) = i
. . | |   by RWO "flip_twice" 0 THEN Auto'
. . | |
. . | .\
. . | . 1. T : Type
. . | . 2. L : T List
. . | . 3. T→T→𝔹
. . | . 4. n : ℕ(||L||-1)
. . | . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . | . 6. ||swap(L;n;n+1)|| = ||L||
. . | . 7. f : ℕ||L||→ℕ||L||
. . | . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
```

```
. . | . 9. i : ℕ||L||
. . | . 10. ℕ||L||
. . | . ⊢  L[i] = L[i]  by Auto
. . |
. . .\
. . . 1. T : Type
. . . 2. L : T List
. . . 3. P : T→T→𝔹
. . . 4. n : ℕ(||L||-1)
. . . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . . 6. ||swap(L;n;n+1)|| = ||L||
. . . 7. f : ℕ||L||→ℕ||L||
. . . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . . 9. i : ℕ||L||
. . . 10. j : ℕ||L||
. . . ⊢  if (f(i)<₂f(j))∧₂P(L[i],swap(L;n;n+1)[(f(j))])→ 1 else 0 fi
. . . |  -if (i<₂j)∧₂P(L[i],L[j])→ 1 else 0 fi
. . . |  =
. . . |  if ((i=₂n)∧₂(j=₂n+1))∧₂P(L[i],L[j])→ -1
. . . |   ; ((i=₂n+1)∧₂(j=₂n))∧₂P(L[i],L[j])→ 1
. . . |  else 0 fi
. . . | by Subst swap(L;n;n+1)[(f(j))] = L[j] 0 THEN Auto'
. . . |
. . . |\
. . . | .....equality .....
. . . | 1. T : Type
. . . | 2. L : T List
. . . | 3. T→T→𝔹
. . . | 4. n : ℕ(||L||-1)
. . . | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . . | 6. ||swap(L;n;n+1)|| = ||L||
. . . | 7. f : ℕ||L||→ℕ||L||
. . . | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . . | 9. ℕ||L||
. . . | 10. j : ℕ||L||
. . . | ⊢  swap(L;n;n+1)[(f(j))] = L[j]
. . . | | by RWO "swap_select" 0 THENA Auto THEN
. . . | |  Subst' f(j)
. . . | |                                                    =
. . . | |                                                    (n, n+1)
. . . | |                                                    (j)
. . . | |     0 THEN EasyHyp
. . . | |
. . . | 1. T : Type
. . . | 2. L : T List
. . . | 3. T→T→𝔹
. . . | 4. n : ℕ(||L||-1)
. . . | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . . | 6. ||swap(L;n;n+1)|| = ||L||
. . . | 7. f : ℕ||L||→ℕ||L||
. . . | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . . | 9. ℕ||L||
. . . | 10. j : ℕ||L||
. . . | ⊢  L[((n, n+1)((n, n+1)(j)))] = L[j]
. . . | | by Subst' (n, n+1)((n, n+1)(j)) = j 0
```

```
. . . | |
. . . | |\
. . . | | 1. T : Type
. . . | | 2. L : T List
. . . | | 3. T→T→𝔹
. . . | | 4. n : ℕ(||L||-1)
. . . | | 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . . | | 6. ||swap(L;n;n+1)|| = ||L||
. . . | | 7. f : ℕ||L||→ℕ||L||
. . . | | 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . . | | 9. ℕ||L||
. . . | | 10. j : ℕ||L||
. . . | | ⊢  (n, n+1)((n, n+1)(j)) = j
. . . | |   by RWO "flip_twice" 0 THEN Auto'
. . . | |
. . . | .\
. . . | . 1. T : Type
. . . | . 2. L : T List
. . . | . 3. T→T→𝔹
. . . | . 4. n : ℕ(||L||-1)
. . . | . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . . | . 6. ||swap(L;n;n+1)|| = ||L||
. . . | . 7. f : ℕ||L||→ℕ||L||
. . . | . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . . | . 9. ℕ||L||
. . . | . 10. j : ℕ||L||
. . . | . ⊢  L[j] = L[j]  by Auto
. . . |
. . . .\
. . . . 1. T : Type
. . . . 2. L : T List
. . . . 3. P : T→T→𝔹
. . . . 4. n : ℕ(||L||-1)
. . . . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . . . 6. ||swap(L;n;n+1)|| = ||L||
. . . . 7. f : ℕ||L||→ℕ||L||
. . . . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . . . 9. i : ℕ||L||
. . . . 10. j : ℕ||L||
. . . . ⊢  if (f(i)<₂f(j))∧₂P(L[i],L[j])→ 1 else 0 fi
. . . . |  -if (i<₂j)∧₂P(L[i],L[j])→ 1 else 0 fi
. . . . |  =
. . . . |  if ((i=₂n)∧₂(j=₂n+1))∧₂P(L[i],L[j])→ -1
. . . . |   ; ((i=₂n+1)∧₂(j=₂n))∧₂P(L[i],L[j])→ 1
. . . . |  else 0 fi
. . . . | by AutoBoolCase i<₂j THEN AutoBoolCase f(i)<₂f(j)
. . . .
. . . . |    THEN MoveToConcl (-1) THEN
. . . . Subst' f(i) = (n, n+1)(i)
. . . . |     0 THEN EasyHyp THEN
. . . . Subst' f(j) = (n, n+1)(j)
. . . . |     0 THEN EasyHyp
. . . . THEN Unfold `flip` 0 THEN Reduce 0 THEN
. . . . |
. . . . (AutoBoolCase i=₂n
```

```
. . . .
. . . . |    THEN MoveToConcl (-1) THEN AutoBoolCase j=₂n
. . . .
. . . . |    THEN MoveToConcl (-1) THEN AutoBoolCase i=₂n+1
. . . .
. . . . |    THEN MoveToConcl (-1) THEN AutoBoolCase j=₂n+1)
. . . . |
. . . . 1. T : Type
. . . . 2. L : T List
. . . . 3. P : T→T→𝔹
. . . . 4. n : ℕ(||L||-1)
. . . . 5. (n, n+1) ∈ ℕ||L||→ℕ||L||
. . . . 6. ||swap(L;n;n+1)|| = ||L||
. . . . 7. f : ℕ||L||→ℕ||L||
. . . . 8. ∀i:ℕ||L||. f(i) = (n, n+1)(i)
. . . . 9. i : ℕ||L||
. . . . 10. j : ℕ||L||
. . . . 11. i<j
. . . . 12. j = n+1
. . . . ⊢  ¬i = n+1
. . . .    ⇒
. . . .    ¬j = n
. . . .    ⇒
. . . .    i = n
. . . .    ⇒
. . . .    n≤n+1
. . . .    ⇒
. . . .    0-if P(L[i],L[j])→ 1 else 0 fi
. . . .    =
. . . .    if P(L[i],L[j])→ -1 else 0 fi
. . . .   by SplitOnConclITE THEN Auto'
END
```