general_length_nth_tl THM

```
⊢  ∀r:ℕ, L:Top List. ||nth_tl(r;L)|| = if r<₂||L||→ ||L||-r else 0 fi
| by InductionOnNat THEN RecUnfold `nth_tl` 0 THEN Reduce 0
|
|\
| ⊢  ∀L:Top List. ||L|| = if 0<₂||L||→ ||L||-0 else 0 fi
|   by Auto THEN SplitOnConclITE
|
.\
. 1. r : ℤ
. 2. 0<r
. 3. ∀L:Top List.
. |  ||nth_tl(r-1;L)|| = if r-1<₂||L||→ ||L||-(r-1) else 0 fi
. ⊢  ∀L:Top List.
. |  ||if r≤₂0→ L else nth_tl(r-1;tl(L)) fi||
. |  =
. |  if r<₂||L||→ ||L||-r else 0 fi
. | by Auto THEN SplitOnConclITE THEN SplitOnConclITE
. |
. |\
. | .....truecase .....
. | 1. r : ℤ
. | 2. 0<r
. | 3. ∀L:Top List.
. |    ||nth_tl(r-1;L)|| = if r-1<₂||L||→ ||L||-(r-1) else 0 fi
. | 4. L : Top List
. | 5. 0<r
. | 6. r<||L||
. | ⊢  ||nth_tl(r-1;tl(L))|| = ||L||-r
. |   by InstHyp [tl(L)] 3
. |      THEN
. |      RWO
. |      Thm* ∀l:A List. ||l||≥1 ⇒ ||tl(l)|| = ||l||-1 ∈ ℤ
. |      {length_tl}
. |      -1
. |      THEN
. |      MoveToConcl -1
. |      THEN
. |      SplitOnConclITE
. |
. .\
. . .....falsecase .....
. . 1. r : ℤ
. . 2. 0<r
. . 3. ∀L:Top List.
. . |  ||nth_tl(r-1;L)|| = if r-1<₂||L||→ ||L||-(r-1) else 0 fi
. . 4. L : Top List
. . 5. 0<r
. . 6. ||L||≤r
. . ⊢  ||nth_tl(r-1;tl(L))|| = 0  by Analyze 4 THEN Reduce 0
. . |
. . |\
. . | 1. r : ℤ
. . | 2. 0<r
```

```
. . | 3. ∀L:Top List.
. . |    ||nth_tl(r-1;L)|| = if r-1<₂||L||→ ||L||-(r-1) else 0 fi
. . | 4. 0<r
. . | 5. ||nil||≤r
. . | ⊢  ||nth_tl(r-1;nil)|| = 0
. . |   by InstHyp [nil] 3 THEN Reduce -1 THEN HypSubst' -1 0
. . |      THEN
. . |      SplitOnConclITE
. . |
. . .\
. . . 1. r : ℤ
. . . 2. 0<r
. . . 3. ∀L:Top List.
. . .    ||nth_tl(r-1;L)|| = if r-1<₂||L||→ ||L||-(r-1) else 0 fi
. . . 4. u : Top
. . . 5. v : Top List
. . . 6. 0<r
. . . 7. ||[u / v]||≤r
. . . ⊢  ||nth_tl(r-1;v)|| = 0
. . .   by InstHyp [v] 3 THEN MoveToConcl -1 THEN SplitOnConclITE
END
```