```
member_upto  THM

⊢  ∀i,j,k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j
| by Assert
|    (∀d,i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j))
|
|\
| .....assertion .....
| ⊢  ∀d,i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| | by InductionOnNat
| |
| |\
| | .....basecase .....
| | ⊢  ∀i,j:ℕ. j-i = 0 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| | | by UnivCD THEN (RecUnfold ‘upto‘ 0)
| | |
| | 1. i : ℕ
| | 2. j : ℕ
| | 3. j-i = 0
| | 4. k : ℕ
| | ⊢  (k ∈ if i<_2j→ [i / upto(i+1;j)] else nil fi) ⟺ i≤k & k<j
| |   by Obvious
| |
| .\
| . .....upcase .....
| . 1. d : ℤ
| . 2. 0<d
| . 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . ⊢  ∀i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | by UnivCD THEN (RecUnfold ‘upto‘ 0) THEN SplitOnConclITE
| . |
| . |\
| . | .....truecase .....
| . | 1. d : ℤ
| . | 2. 0<d
| . | 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | 4. i : ℕ
| . | 5. j : ℕ
| . | 6. j-i = d
| . | 7. k : ℕ
| . | 8. i<j
| . | ⊢  (k ∈ [i / upto(i+1;j)]) ⟺ i≤k & k<j  by InstHyp [i+1;j;k] 3
| . | |
| . | 1. d : ℤ
| . | 2. 0<d
| . | 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | 4. i : ℕ
| . | 5. j : ℕ
| . | 6. j-i = d
| . | 7. k : ℕ
| . | 8. i<j
| . | 9. (k ∈ upto(i+1;j)) ⟺ i+1≤k & k<j
| . | ⊢  (k ∈ [i / upto(i+1;j)]) ⟺ i≤k & k<j
| . | | by (Decide (k = i))
| . | |    THEN
```

```
| . | |     (RWW
| . | |      Thm* ∀l:T List, a,x:T. (x ∈ [a / l]) ⟺ x = a ∨ (x ∈ l)
| . | |      {cons_member}
| . | |      0)
| . | |
| . | |\
| . | | 1. d : ℤ
| . | | 2. 0<d
| . | | 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | | 4. i : ℕ
| . | | 5. j : ℕ
| . | | 6. j-i = d
| . | | 7. k : ℕ
| . | | 8. i<j
| . | | 9. (k ∈ upto(i+1;j)) ⇒ i+1≤k & k<j
| . | | 10. (k ∈ upto(i+1;j)) ⇐ (i+1≤k & k<j)
| . | | 11. ¬k = i
| . | | 12. k = i ∨ (k ∈ upto(i+1;j))
| . | | ⊢  i≤k  by Assert (i+1≤k)
| . | | |
| . | | |\
| . | | | .....assertion .....
| . | | | 1. d : ℤ
| . | | | 2. 0<d
| . | | | 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | | | 4. i : ℕ
| . | | | 5. j : ℕ
| . | | | 6. j-i = d
| . | | | 7. k : ℕ
| . | | | 8. i<j
| . | | | 9. (k ∈ upto(i+1;j)) ⇒ i+1≤k & k<j
| . | | | 10. (k ∈ upto(i+1;j)) ⇐ (i+1≤k & k<j)
| . | | | 11. ¬k = i
| . | | | 12. k = i ∨ (k ∈ upto(i+1;j))
| . | | | ⊢  i+1≤k  by Obvious
| . | | |
| . | | .\
| . | | . 1. d : ℤ
| . | | . 2. 0<d
| . | | . 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | | . 4. i : ℕ
| . | | . 5. j : ℕ
| . | | . 6. j-i = d
| . | | . 7. k : ℕ
| . | | . 8. i<j
| . | | . 9. (k ∈ upto(i+1;j)) ⇒ i+1≤k & k<j
| . | | . 10. (k ∈ upto(i+1;j)) ⇐ (i+1≤k & k<j)
| . | | . 11. ¬k = i
| . | | . 12. k = i ∨ (k ∈ upto(i+1;j))
| . | | . 13. i+1≤k
| . | | . ⊢  i≤k  by Auto
| . | |
| . | |\
| . | | 1. d : ℤ
| . | | 2. 0<d
```

```
| . | | 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | | 4. i : ℕ
| . | | 5. j : ℕ
| . | | 6. j-i = d
| . | | 7. k : ℕ
| . | | 8. i<j
| . | | 9. (k ∈ upto(i+1;j)) ⇒ i+1≤k & k<j
| . | | 10. (k ∈ upto(i+1;j)) ⇐ (i+1≤k & k<j)
| . | | 11. ¬k = i
| . | | 12. k = i ∨ (k ∈ upto(i+1;j))
| . | | ⊢  k<j  by Obvious
| . | |
| . | .\
| . | . 1. d : ℤ
| . | . 2. 0<d
| . | . 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . | . 4. i : ℕ
| . | . 5. j : ℕ
| . | . 6. j-i = d
| . | . 7. k : ℕ
| . | . 8. i<j
| . | . 9. (k ∈ upto(i+1;j)) ⇒ i+1≤k & k<j
| . | . 10. (k ∈ upto(i+1;j)) ⇐ (i+1≤k & k<j)
| . | . 11. ¬k = i
| . | . 12. i≤k
| . | . 13. k<j
| . | . ⊢  k = i ∨ (k ∈ upto(i+1;j))  by Obvious
| . |
| . .\
| . . .....falsecase .....
| . . 1. d : ℤ
| . . 2. 0<d
| . . 3. ∀i,j:ℕ. j-i = d-1 ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
| . . 4. i : ℕ
| . . 5. j : ℕ
| . . 6. j-i = d
| . . 7. k : ℕ
| . . 8. j≤i
| . . ⊢  (k ∈ nil) ⟺ i≤k & k<j  by Obvious
|
.\
. 1. ∀d,i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
. ⊢  ∀i,j,k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j
. | by UnivCD THEN (Decide (i≤j))
. |
. |\
. | 1. ∀d,i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
. | 2. i : ℕ
. | 3. j : ℕ
. | 4. k : ℕ
. | 5. i≤j
. | ⊢  (k ∈ upto(i;j)) ⟺ i≤k & k<j
. |   by (InstHyp [j-i;i;j;k] 1) THEN Trivial
. |
. .\
```

```
. . 1. ∀d,i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
. . 2. i : ℕ
. . 3. j : ℕ
. . 4. k : ℕ
. . 5. ¬i≤j
. . ⊢  (k ∈ upto(i;j)) ⟺ i≤k & k<j
. . | by (RecUnfold `upto` 0) THEN SplitOnConclITE
. . |
. . |\
. . | .....truecase .....
. . | 1. ∀d,i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
. . | 2. i : ℕ
. . | 3. j : ℕ
. . | 4. k : ℕ
. . | 5. ¬i≤j
. . | 6. i<j
. . | ⊢  (k ∈ [i / upto(i+1;j)]) ⟺ i≤k & k<j  by Obvious
. . |
. . .\
. . . .....falsecase .....
. . . 1. ∀d,i,j:ℕ. j-i = d ⇒ (∀k:ℕ. (k ∈ upto(i;j)) ⟺ i≤k & k<j)
. . . 2. i : ℕ
. . . 3. j : ℕ
. . . 4. k : ℕ
. . . 5. ¬i≤j
. . . 6. j≤i
. . . ⊢  (k ∈ nil) ⟺ i≤k & k<j  by Obvious
END
```