```
formula_and_right_sat  THM

⊢  ∀hyp,M,N:Formula List, q,r:Formula, a:Assignment.
|  a |= <hyp,q.(M @ N)> & a |= <hyp,r.(M @ N)>
|  ⟺
|  a |= <hyp,M @ (q⌜∧⌝r.N)>  by UnivCD THEN Analyze 0 THEN Analyze 0
|
|\
| 1. hyp : Formula List
| 2. M : Formula List
| 3. N : Formula List
| 4. q : Formula
| 5. r : Formula
| 6. a : Assignment
| 7. a |= <hyp,q.(M @ N)> & a |= <hyp,r.(M @ N)>
| ⊢  a |= <hyp,M @ (q⌜∧⌝r.N)>  by Analyze -1
| |
| 7. a |= <hyp,q.(M @ N)>
| 8. a |= <hyp,r.(M @ N)>
| ⊢
| | by OnHyps [-2;-1;0]
| |    (λi.
| |     UnfoldTopAb i THEN AbReduce i
| |     THEN
| |     Rewrite
| |     (HigherC
| |      (LemmaC
| |       Thm* ∀P:(T→Prop), L,M:T List.
| |            ∃x∈(L @ M).P(x) ⟺ ∃x∈L.P(x) ∨ ∃x∈M.P(x)
| |       {list_exists_append_lemma}))
| |     i
| |     THEN
| |     AbReduce i)
| |
| 7. ∃F∈hyp.a |≠ F ∨ a |= q  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
| 8. ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
| ⊢  ∃F∈hyp.a |≠ F ∨ ∃F∈M.a |= F  ∨ a |= q⌜∧⌝r  ∨ ∃F∈N.a |= F
| | by Rewrite
| |    (HigherC
| |     (LemmaC
| |      Thm* ∀a:Assignment, q,r:Formula.
| |           a |= q⌜∧⌝r  ⟺ a |= q  & a |= r
| |      {formula_and_sat_lemma}))
| |    0
| |
| ⊢  ∃F∈hyp.a |≠ F ∨ ∃F∈M.a |= F  ∨ a |= q  & a |= r  ∨ ∃F∈N.a |= F
| | by OnHyps [-2] RepeatOrHD
| |
| |\
| | 7. ∃F∈hyp.a |≠ F
| | 8. ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
| | ⊢   by Choose [1] THEN Trivial
| |
| |\
| | 7. a |= q
```

```
| | 8. ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
| | ⊢   by RepeatOrHD -1
| | |
| | |\
| | | 8. ∃F∈hyp.a |≠ F
| | | ⊢   by Choose [1] THEN Trivial
| | |
| | |\
| | | 8. a |= r
| | | ⊢   by Choose [2;2;1]
| | |
| | |\
| | | 8. ∃F∈M.a |= F
| | | ⊢   by Choose [2;1]
| | |
| | .\
| | . 8. ∃F∈N.a |= F
| | . ⊢   by Choose [2;2;2]
| |
| |\
| | 7. ∃F∈M.a |= F
| | 8. ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
| | ⊢   by Choose [2;1] THEN Trivial
| |
| .\
| . 7. ∃F∈N.a |= F
| . 8. ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
| . ⊢   by Choose [2;2;2]
|
.\
. 1. hyp : Formula List
. 2. M : Formula List
. 3. N : Formula List
. 4. q : Formula
. 5. r : Formula
. 6. a : Assignment
. 7. a |= <hyp,M @ (q⌈∧⌉r.N)>
. ⊢  a |= <hyp,q.(M @ N)> & a |= <hyp,r.(M @ N)>
. | by OnHyps [-1]
. |    (λi.
. |     UnfoldTopAb i THEN AbReduce i
. |     THEN
. |     Rewrite
. |     (HigherC
. |      (LemmaC
. |       Thm* ∀P:(T→Prop), L,M:T List.
. |            ∃x∈(L @ M).P(x) ⟺ ∃x∈L.P(x) ∨ ∃x∈M.P(x)
. |       {list_exists_append_lemma}))
. |     i
. |     THEN
. |     AbReduce i)
. |
. 7. ∃F∈hyp.a |≠ F ∨ ∃F∈M.a |= F  ∨ a |= q⌈∧⌉r  ∨ ∃F∈N.a |= F
. ⊢
. | by Rewrite
```

```
. |    (HigherC
. |     (LemmaC
. |      Thm* ∀a:Assignment, q,r:Formula.
. |           a |= q⌈∧⌉r  ⟺ a |= q  & a |= r
. |      {formula_and_sat_lemma}))
. |    -1
. |
. 7. ∃F∈hyp.a |≠ F ∨ ∃F∈M.a |= F  ∨ a |= q  & a |= r  ∨ ∃F∈N.a |= F
. ⊢
. | by Analyze 0
. |    THEN
. |    OnHyps [0]
. |    (λi.
. |     UnfoldTopAb i THEN AbReduce i
. |     THEN
. |     Rewrite
. |     (HigherC
. |      (LemmaC
. |       Thm* ∀P:(T→Prop), L,M:T List.
. |            ∃x∈(L @ M).P(x) ⟺ ∃x∈L.P(x) ∨ ∃x∈M.P(x)
. |       {list_exists_append_lemma}))
. |     i
. |     THEN
. |     AbReduce i)
. |    THEN
. |    RepeatOrHD -1
. |
. |\
. | 7. ∃F∈hyp.a |≠ F
. | ⊢  ∃F∈hyp.a |≠ F ∨ a |= q  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
. |   by Choose [1]
. |
. |\
. | 7. ∃F∈M.a |= F
. | ⊢  ∃F∈hyp.a |≠ F ∨ a |= q  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
. |   by Choose [2;2;1] THEN Trivial
. |
. |\
. | 7. a |= q  & a |= r
. | ⊢  ∃F∈hyp.a |≠ F ∨ a |= q  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
. |   by Analyze -1 THEN Choose [2;1] THEN Trivial
. |
. |\
. | 7. ∃F∈N.a |= F
. | ⊢  ∃F∈hyp.a |≠ F ∨ a |= q  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
. |   by Choose [2;2;2] THEN Trivial
. |
. |\
. | 7. ∃F∈hyp.a |≠ F
. | ⊢  ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
. |   by Choose [1] THEN Trivial
. |
. |\
. | 7. ∃F∈M.a |= F
. | ⊢  ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
```

```
. |   by Choose [2;2;1] THEN Trivial
. |
. |\
. | 7. a |= q  & a |= r
. | ⊢  ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
. |   by Analyze -1 THEN Choose [2;1] THEN Trivial
. |
. .\
. . 7. ∃F∈N.a |= F
. . ⊢  ∃F∈hyp.a |≠ F ∨ a |= r  ∨ ∃F∈M.a |= F  ∨ ∃F∈N.a |= F
. .   by Choose [2;2;2] THEN Trivial
END
```