```
formula_imp_left_sat  THM

⊢  ∀concl,M,N:Formula List, q,r:Formula, a:Assignment.
|  a |= <r.(M @ N),concl> & a |= <M @ N,q.concl>
|  ⟺
|  a |= <M @ (q⌜⇒⌝r.N),concl>
| by UnivCD THEN Analyze 0 THEN Analyze 0
|
|\
| 1. concl : Formula List
| 2. M : Formula List
| 3. N : Formula List
| 4. q : Formula
| 5. r : Formula
| 6. a : Assignment
| 7. a |= <r.(M @ N),concl> & a |= <M @ N,q.concl>
| ⊢  a |= <M @ (q⌜⇒⌝r.N),concl>  by Analyze -1
| |
| 7. a |= <r.(M @ N),concl>
| 8. a |= <M @ N,q.concl>
| ⊢
| | by OnHyps [-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)
| |
| 8. ∃F∈M.a |≠ F ∨ ∃F∈N.a |≠ F ∨ a |= q  ∨ ∃F∈concl.a |= F
| ⊢  ∃F∈M.a |≠ F ∨ a |≠ q⌜⇒⌝r ∨ ∃F∈N.a |≠ F ∨ ∃F∈concl.a |= F
| | by Analyze -1 THEN Analyze -1
| |
| |\
| | 8. ∃F∈M.a |≠ F
| | ⊢   by Choose [1;1] THEN Trivial
| |
| |\
| | 8. ∃F∈N.a |≠ F
| | ⊢   by Choose [1;2;2] THEN Trivial
| |
| |\
| | 8. a |= q
| | ⊢
| | | by OnHyps [-2]
| | |    (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. a |≠ r ∨ ∃F∈M.a |≠ F ∨ ∃F∈N.a |≠ F ∨ ∃F∈concl.a |= F
| | 8. a |= q
| | ⊢   by Analyze -2
| | |
| | |\
| | | 7. a |≠ r ∨ ∃F∈M.a |≠ F ∨ ∃F∈N.a |≠ F
| | | 8. a |= q
| | | ⊢   by Analyze -2
| | | |
| | | |\
| | | | 7. a |≠ r
| | | | 8. a |= q
| | | | ⊢   by Choose [1;2;1]
| | | | |
| | | | ⊢  a |≠ q⌈⇒⌉r
| | | |   by BackThru
| | | |      Thm* ∀a:Assignment, q,r:Formula.
| | | |           a |≠ q⌈⇒⌉r ⟺ a |= q  & a |≠ r
| | | |      {formula_imp_falsifiable_lemma}
| | | |
| | | .\
| | | . 7. ∃F∈M.a |≠ F ∨ ∃F∈N.a |≠ F
| | | . 8. a |= q
| | | . ⊢   by Analyze -2
| | | . |
| | | . |\
| | | . | 7. ∃F∈M.a |≠ F
| | | . | 8. a |= q
| | | . | ⊢   by Choose [1;1] THEN Trivial
| | | . |
| | | . .\
| | | . . 7. ∃F∈N.a |≠ F
| | | . . 8. a |= q
| | | . . ⊢   by Choose [1;2;2] THEN Trivial
| | |
| | .\
| | . 7. ∃F∈concl.a |= F
| | . 8. a |= q
| | . ⊢   by Choose [2] THEN Trivial
| |
| .\
| . 8. ∃F∈concl.a |= F
| . ⊢   by Choose [2] THEN Trivial
|
.\
. 1. concl : Formula List
```

```
. 2. M : Formula List
. 3. N : Formula List
. 4. q : Formula
. 5. r : Formula
. 6. a : Assignment
. 7. a |= <M @ (q⌜⇒⌝r.N),concl>
. ⊢  a |= <r.(M @ N),concl> & a |= <M @ N,q.concl>  by Analyze 0
. |
. |\
. | ⊢  a |= <r.(M @ N),concl>
. | | by OnHyps [-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∈M.a |≠ F ∨ a |≠ q⌜⇒⌝r ∨ ∃F∈N.a |≠ F ∨ ∃F∈concl.a |= F
. | ⊢  a |≠ r ∨ ∃F∈M.a |≠ F ∨ ∃F∈N.a |≠ F ∨ ∃F∈concl.a |= F
. | | by Analyze -1
. | |
. | |\
. | | 7. ∃F∈M.a |≠ F ∨ a |≠ q⌜⇒⌝r ∨ ∃F∈N.a |≠ F
. | | ⊢   by Analyze -1
. | | |
. | | |\
. | | | 7. ∃F∈M.a |≠ F
. | | | ⊢   by Choose [1;2;1] THEN Trivial
. | | |
. | | .\
. | | . 7. a |≠ q⌜⇒⌝r ∨ ∃F∈N.a |≠ F
. | | . ⊢   by Analyze -1
. | | . |
. | | . |\
. | | . | 7. a |≠ q⌜⇒⌝r
. | | . | ⊢
. | | . | | by Sel 2
. | | . | |    (FwdThru
. | | . | |     Thm* ∀a:Assignment, q,r:Formula.
. | | . | |          a |≠ q⌜⇒⌝r ⟺ a |= q  & a |≠ r
. | | . | |     {formula_imp_falsifiable_lemma}
. | | . | |     [-1])
. | | . | |
. | | . | 8. a |≠ r
. | | . | ⊢   by Choose [1;1]
. | | . |
. | | . .\
. | | . . 7. ∃F∈N.a |≠ F
```

```
. | | . . ⊢   by Choose [1;2;2] THEN Trivial
. | |
. | .\
. | . 7. ∃F∈concl.a |= F
. | . ⊢   by Choose [2] THEN Trivial
. |
. .\
. . ⊢  a |= <M @ N,q.concl>
. . | by OnHyps [-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∈M.a |≠ F ∨ a |≠ q⌜⇒⌝r ∨ ∃F∈N.a |≠ F ∨ ∃F∈concl.a |= F
. . ⊢  ∃F∈M.a |≠ F ∨ ∃F∈N.a |≠ F ∨ a |= q  ∨ ∃F∈concl.a |= F
. . | by Analyze -1
. . |
. . |\
. . | 7. ∃F∈M.a |≠ F ∨ a |≠ q⌜⇒⌝r ∨ ∃F∈N.a |≠ F
. . | ⊢   by Analyze -1
. . | |
. . | |\
. . | | 7. ∃F∈M.a |≠ F
. . | | ⊢   by Choose [1;1] THEN Trivial
. . | |
. . | .\
. . | . 7. a |≠ q⌜⇒⌝r ∨ ∃F∈N.a |≠ F
. . | . ⊢   by Analyze -1
. . | . |
. . | . |\
. . | . | 7. a |≠ q⌜⇒⌝r
. . | . | ⊢
. . | . | | by Sel 1
. . | . | |    (FwdThru
. . | . | |     Thm* ∀a:Assignment, q,r:Formula.
. . | . | |          a |≠ q⌜⇒⌝r ⟺ a |= q  & a |≠ r
. . | . | |     {formula_imp_falsifiable_lemma}
. . | . | |     [-1])
. . | . | |
. . | . | 8. a |= q
. . | . | ⊢   by Choose [2;1]
. . | . |
. . | . .\
. . | . . 7. ∃F∈N.a |≠ F
. . | . . ⊢   by Choose [1;2] THEN Trivial
. . |
```

```
. . .\
. . . 7. ∃F∈concl.a |= F
. . . ⊢   by Choose [2;2] THEN Trivial
END
```