- Output voor probleem 9 uit Pelletier's lijst.

========== Proof: ==========

With ()   and AXIOM   : 10. p,p |- p,p
With (10)   and Left- ~ : 9. ~p,p,p |- p
With ()   and AXIOM   : 11. p,p,q |- p
With (9,11) and Left- ∨ : 8. (~p ∨ q),p,p |- p
With ()   and AXIOM   : 14. p,q |- p,p
With (14)   and Left- ~ : 13. ~p,p,q |- p
With ()   and AXIOM   : 15. p,q,q |- p
With (13,15) and Left- ∨ : 12. (~p ∨ q),p,q |- p
With (8,12) and Left- ∨ : 7. (p ∨ q),(~p ∨ q),p |- p
With (7)   and Left- ~ : 6. ~p,(p ∨ q),(~p ∨ q),p |-
With ()   and AXIOM   : 21. p |- p,q,p
With (21)   and Left- ~ : 20. ~p,p |- p,q
With (20)   and Left- ~ : 19. ~q,p,~p |- p
With ()   and AXIOM   : 23. p,q |- p,q
With (23)   and Left- ~ : 22. ~q,p,q |- p
With (19,22) and Left- ∨ : 18. (~p ∨ q),~q,p |- p
With ()   and AXIOM   : 27. q |- p,q,p
With (27)   and Left- ~ : 26. ~p,q |- p,q
With (26)   and Left- ~ : 25. ~q,q,~p |- p
With ()   and AXIOM   : 29. q,q |- p,q
With (29)   and Left- ~ : 28. ~q,q,q |- p
With (25,28) and Left- ∨ : 24. (~p ∨ q),~q,q |- p
With (18,24) and Left- ∨ : 17. (p ∨ q),(~p ∨ q),~q |- p
With (17)   and Left- ~ : 16. ~p,(p ∨ q),(~p ∨ q),~q |-
With (6,16) and Left- ∨ : 5. (p ∨ ~q),~p,(p ∨ q),(~p ∨ q) |-
With (5)   and Left- ∧ : 4. ((p ∨ q) ∧ (~p ∨ q)),(p ∨ ~q),~p |-
With ()   and AXIOM   : 36. p,p |- q,p
With (36)   and Left- ~ : 35. ~p,p,p |- q
With ()   and AXIOM   : 37. p,p,q |- q
With (35,37) and Left- ∨ : 34. (~p ∨ q),p,p |- q
With ()   and AXIOM   : 40. p,q |- q,p
With (40)   and Left- ~ : 39. ~p,p,q |- q
With ()   and AXIOM   : 41. p,q,q |- q
With (39,41) and Left- ∨ : 38. (~p ∨ q),p,q |- q
With (34,38) and Left- ∨ : 33. (p ∨ q),(~p ∨ q),p |- q
With (33)   and Left- ~ : 32. ~q,(p ∨ q),(~p ∨ q),p |-
With ()   and AXIOM   : 47. p |- q,q,p
With (47)   and Left- ~ : 46. ~p,p |- q,q
With (46)   and Left- ~ : 45. ~q,p,~p |- q
With ()   and AXIOM   : 49. p,q |- q,q
With (49)   and Left- ~ : 48. ~q,p,q |- q
With (45,48) and Left- ∨ : 44. (~p ∨ q),~q,p |- q
With ()   and AXIOM   : 53. q |- q,q,p
With (53)   and Left- ~ : 52. ~p,q |- q,q
With (52)   and Left- ~ : 51. ~q,q,~p |- q
With ()   and AXIOM   : 55. q,q |- q,q
With (55)   and Left- ~ : 54. ~q,q,q |- q
With (51,54) and Left- ∨ : 50. (~p ∨ q),~q,q |- q
With (44,50) and Left- ∨ : 43. (p ∨ q),(~p ∨ q),~q |- q
With (43)   and Left- ~ : 42. ~q,(p ∨ q),(~p ∨ q),~q |-
With (32,42) and Left- ∨ : 31. (p ∨ ~q),~q,(p ∨ q),(~p ∨ q) |-

With (31) and Left- /\ : 30. ((p \/ q) /\ (~p \/ q)),(p \/ ~q),~q |-
With (4,30) and Left- \/ : 3. (~p \/ ~q),((p \/ q) /\ (~p \/ q)),(p \/ ~q) |-
With (3) and Left- /\ : 2. (((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q)),(~p \/ ~q) |-
With (2) and Right- ~ : 1. (((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q)) |- ~(~p \/ ~q)
With (1) and Right- => : 0. |- ((((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q)) => ~(~p \/ ~q))

========== QED ==

- Listing van een complete trace voor probleem 9 uit Pelletier's lijst.1

Adding initial proposition to RHS... (Trying to find countermodel)
Fireing up ATP:
Performing iteration with:
LHS:
RHS: ((((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q)) => ~(~p \/ ~q))
Non-atomic proposition found on RHS...
Iterating on the right-hand side...
0. Right- => on ((((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q)) => ~(~p \/ ~q)) with
LHS:
RHS:
Performing iteration with:
LHS: (((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q))
RHS: ~(~p \/ ~q)
Non-atomic proposition found on RHS...
Iterating on the right-hand side...
1. Right- ~ on ~(~p \/ ~q) with
LHS: (((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q))
RHS:
Performing iteration with:
LHS: (((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q)),(~p \/ ~q)
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
2. Left- /\ on (((p \/ q) /\ (~p \/ q)) /\ (p \/ ~q)) with
LHS: (~p \/ ~q)
RHS:
Performing iteration with:
LHS: (~p \/ ~q),((p \/ q) /\ (~p \/ q)),(p \/ ~q)
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
3. Left- \/ on (~p \/ ~q) with
LHS: ((p \/ q) /\ (~p \/ q)),(p \/ ~q)
RHS:
left split:
Performing iteration with:
LHS: ((p \/ q) /\ (~p \/ q)),(p \/ ~q),~p
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
4. Left- /\ on ((p \/ q) /\ (~p \/ q)) with
LHS: (p \/ ~q),~p
RHS:
Performing iteration with:
LHS: (p \/ ~q),~p,(p \/ q),(~p \/ q)
RHS:

Non-atomic proposition found on LHS...
Iterating on the left-hand side...
5. Left- \/ on (p \/ ~q) with
LHS: ~p,(p \/ q),(~p \/ q)
RHS:
left split:
Performing iteration with:
LHS: ~p,(p \/ q),(~p \/ q),p
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
6. Left- ~ on ~p with
LHS: (p \/ q),(~p \/ q),p
RHS:
Performing iteration with:
LHS: (p \/ q),(~p \/ q),p
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
7. Left- \/ on (p \/ q) with
LHS: (~p \/ q),p
RHS: p
left split:
Performing iteration with:
LHS: (~p \/ q),p,p
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
8. Left- \/ on (~p \/ q) with
LHS: p,p
RHS: p
left split:
Performing iteration with:
LHS: p,p,~p
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
9. Left- ~ on ~p with
LHS: p,p
RHS: p
Performing iteration with:
LHS: p,p
RHS: p,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,p]
[p,p]
Checking if p is contained in
[p,p]
True
Sequent is Axiom!
No contradiction.
Close branch.
9. yields:
LHS: [p,p]
RHS: [p,p]

8. yields:

LHS: [p,p,~p]
RHS: [p]

right split:
Performing iteration with:
LHS: p,p,q
RHS: p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,p,q]
[p]
Checking if p is contained in
[p]
True
Sequent is Axiom!
No contradiction.
Close branch.
8. yields:
LHS: [p,p,q]
RHS: [p]

7. yields:
LHS: [(~p ∨ q),p,p]
RHS: [p]

right split:
Performing iteration with:
LHS: (~p ∨ q),p,q
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
12. Left- ∨ on (~p ∨ q) with
LHS: p,q
RHS: p
left split:
Performing iteration with:
LHS: p,q,~p
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
13. Left- ~ on ~p with
LHS: p,q
RHS: p
Performing iteration with:
LHS: p,q
RHS: p,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,q]
[p,p]
Checking if p is contained in
[p,p]
True
Sequent is Axiom!
No contradiction.
Close branch.
13. yields:
LHS: [p,q]

RHS: [p,p]

12. yields:
LHS: [p,q,~p]
RHS: [p]

right split:
Performing iteration with:
LHS: p,q,q
RHS: p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,q,q]
[p]
Checking if p is contained in
[p]
True
Sequent is Axiom!
No contradiction.
Close branch.
12. yields:
LHS: [p,q,q]
RHS: [p]

7. yields:
LHS: [(~p \/ q),p,q]
RHS: [p]

6. yields:
LHS: [(p \/ q),(~p \/ q),p]
RHS: [p]

5. yields:
LHS: [~p,(p \/ q),(~p \/ q),p]
RHS: []

right split:
Performing iteration with:
LHS: ~p,(p \/ q),(~p \/ q),~q
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
16. Left- ~ on ~p with
LHS: (p \/ q),(~p \/ q),~q
RHS:
Performing iteration with:
LHS: (p \/ q),(~p \/ q),~q
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
17. Left- \/ on (p \/ q) with
LHS: (~p \/ q),~q
RHS: p
left split:
Performing iteration with:
LHS: (~p \/ q),~q,p
RHS: p
Non-atomic proposition found on LHS...

Iterating on the left-hand side...
18. Left- \/ on (~p \/ q) with
LHS: ~q,p
RHS: p
left split:
Performing iteration with:
LHS: ~q,p,~p
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
19. Left- ~ on ~q with
LHS: p,~p
RHS: p
Performing iteration with:
LHS: p,~p
RHS: p,q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
20. Left- ~ on ~p with
LHS: p
RHS: p,q
Performing iteration with:
LHS: p
RHS: p,q,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p]
[p,q,p]
Checking if p is contained in
[p,q,p]
True
Sequent is Axiom!
No contradiction.
Close branch.
20. yields:
LHS: [p]
RHS: [p,q,p]

19. yields:
LHS: [p,~p]
RHS: [p,q]

18. yields:
LHS: [~q,p,~p]
RHS: [p]

right split:
Performing iteration with:
LHS: ~q,p,q
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
22. Left- ~ on ~q with
LHS: p,q
RHS: p
Performing iteration with:
LHS: p,q
RHS: p,q

LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,q]
[p,q]
Checking if p is contained in
[p,q]
True
Sequent is Axiom!
No contradiction.
Close branch.
22. yields:
LHS: [p,q]
RHS: [p,q]

18. yields:
LHS: [~q,p,q]
RHS: [p]

17. yields:
LHS: [(~p \/ q),~q,p]
RHS: [p]

right split:
Performing iteration with:
LHS: (~p \/ q),~q,q
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
24. Left- \/ on (~p \/ q) with
LHS: ~q,q
RHS: p
left split:
Performing iteration with:
LHS: ~q,q,~p
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
25. Left- ~ on ~q with
LHS: q,~p
RHS: p
Performing iteration with:
LHS: q,~p
RHS: p,q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
26. Left- ~ on ~p with
LHS: q
RHS: p,q
Performing iteration with:
LHS: q
RHS: p,q,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[q]
[p,q,p]
Checking if q is contained in
[p,q,p]
True

Sequent is Axiom!
No contradiction.
Close branch.
26. yields:
LHS: [q]
RHS: [p,q,p]

25. yields:
LHS: [q,~p]
RHS: [p,q]

24. yields:
LHS: [~q,q,~p]
RHS: [p]

right split:
Performing iteration with:
LHS: ~q,q,q
RHS: p
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
28. Left- ~ on ~q with
LHS: q,q
RHS: p
Performing iteration with:
LHS: q,q
RHS: p,q
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[q,q]
[p,q]
Checking if q is contained in
[p,q]
True
Sequent is Axiom!
No contradiction.
Close branch.
28. yields:
LHS: [q,q]
RHS: [p,q]

24. yields:
LHS: [~q,q,q]
RHS: [p]

17. yields:
LHS: [(~p \/ q),~q,q]
RHS: [p]

16. yields:
LHS: [(p \/ q),(~p \/ q),~q]
RHS: [p]

5. yields:
LHS: [~p,(p \/ q),(~p \/ q),~q]
RHS: []

4. vields:

LHS: [(p ∨ ~q),~p,(p ∨ q),(~p ∨ q)]
RHS: []

3. yields:
LHS: [((p ∨ q) ∧ (~p ∨ q)),(p ∨ ~q),~p]
RHS: []

right split:
Performing iteration with:
LHS: ((p ∨ q) ∧ (~p ∨ q)),(p ∨ ~q),~q
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
30. Left- ∧ on ((p ∨ q) ∧ (~p ∨ q)) with
LHS: (p ∨ ~q),~q
RHS:
Performing iteration with:
LHS: (p ∨ ~q),~q,(p ∨ q),(~p ∨ q)
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
31. Left- ∨ on (p ∨ ~q) with
LHS: ~q,(p ∨ q),(~p ∨ q)
RHS:
left split:
Performing iteration with:
LHS: ~q,(p ∨ q),(~p ∨ q),p
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
32. Left- ~ on ~q with
LHS: (p ∨ q),(~p ∨ q),p
RHS:
Performing iteration with:
LHS: (p ∨ q),(~p ∨ q),p
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
33. Left- ∨ on (p ∨ q) with
LHS: (~p ∨ q),p
RHS: q
left split:
Performing iteration with:
LHS: (~p ∨ q),p,p
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
34. Left- ∨ on (~p ∨ q) with
LHS: p,p
RHS: q
left split:
Performing iteration with:
LHS: p,p,~p
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
35. Left- ~ on ~p with
LHS: p,p

RHS: q
Performing iteration with:
LHS: p,p
RHS: q,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,p]
[q,p]
Checking if p is contained in
[q,p]
True
Sequent is Axiom!
No contradiction.
Close branch.
35. yields:
LHS: [p,p]
RHS: [q,p]

34. yields:
LHS: [p,p,~p]
RHS: [q]

right split:
Performing iteration with:
LHS: p,p,q
RHS: q
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,p,q]
[q]
Checking if p is contained in
[q]
False
Checking if p is contained in
[q]
False
Checking if q is contained in
[q]
True
Sequent is Axiom!
No contradiction.
Close branch.
34. yields:
LHS: [p,p,q]
RHS: [q]

33. yields:
LHS: [(~p ∨ q),p,p]
RHS: [q]

right split:
Performing iteration with:
LHS: (~p ∨ q),p,q
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
38. Left- ∨ on (~p ∨ q) with
LHS: p,q

RHS: q
left split:
Performing iteration with:
LHS: p,q,~p
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
39. Left- ~ on ~p with
LHS: p,q
RHS: q
Performing iteration with:
LHS: p,q
RHS: q,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,q]
[q,p]
Checking if p is contained in
[q,p]
True
Sequent is Axiom!
No contradiction.
Close branch.
39. yields:
LHS: [p,q]
RHS: [q,p]

38. yields:
LHS: [p,q,~p]
RHS: [q]

right split:
Performing iteration with:
LHS: p,q,q
RHS: q
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,q,q]
[q]
Checking if p is contained in
[q]
False
Checking if q is contained in
[q]
True
Sequent is Axiom!
No contradiction.
Close branch.
38. yields:
LHS: [p,q,q]
RHS: [q]

33. yields:
LHS: [(~p \/ q),p,q]
RHS: [q]

32. yields:
LHS: [(p \/ q),(~p \/ q),p]

RHS: [q]

31. yields:
LHS: [~q,(p \/ q),(~p \/ q),p]
RHS: []

right split:
Performing iteration with:
LHS: ~q,(p \/ q),(~p \/ q),~q
RHS:
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
42. Left- ~ on ~q with
LHS: (p \/ q),(~p \/ q),~q
RHS:
Performing iteration with:
LHS: (p \/ q),(~p \/ q),~q
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
43. Left- \/ on (p \/ q) with
LHS: (~p \/ q),~q
RHS: q
left split:
Performing iteration with:
LHS: (~p \/ q),~q,p
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
44. Left- \/ on (~p \/ q) with
LHS: ~q,p
RHS: q
left split:
Performing iteration with:
LHS: ~q,p,~p
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
45. Left- ~ on ~q with
LHS: p,~p
RHS: q
Performing iteration with:
LHS: p,~p
RHS: q,q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
46. Left- ~ on ~p with
LHS: p
RHS: q,q
Performing iteration with:
LHS: p
RHS: q,q,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p]
[q,q,p]
Checking if p is contained in
[q,q,p]

True
Sequent is Axiom!
No contradiction.
Close branch.
46. yields:
LHS: [p]
RHS: [q,q,p]

45. yields:
LHS: [p,~p]
RHS: [q,q]

44. yields:
LHS: [~q,p,~p]
RHS: [q]

right split:
Performing iteration with:
LHS: ~q,p,q
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
48. Left- ~ on ~q with
LHS: p,q
RHS: q
Performing iteration with:
LHS: p,q
RHS: q,q
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[p,q]
[q,q]
Checking if p is contained in
[q,q]
False
Checking if q is contained in
[q,q]
True
Sequent is Axiom!
No contradiction.
Close branch.
48. yields:
LHS: [p,q]
RHS: [q,q]

44. yields:
LHS: [~q,p,q]
RHS: [q]

43. yields:
LHS: [(~p \/ q),~q,p]
RHS: [q]

right split:
Performing iteration with:
LHS: (~p \/ q),~q,q
RHS: q
Non-atomic proposition found on LHS

Iterating on the left-hand side...
50. Left- ∨ on (~p ∨ q) with
LHS: ~q,q
RHS: q
left split:
Performing iteration with:
LHS: ~q,q,~p
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
51. Left- ~ on ~q with
LHS: q,~p
RHS: q
Performing iteration with:
LHS: q,~p
RHS: q,q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
52. Left- ~ on ~p with
LHS: q
RHS: q,q
Performing iteration with:
LHS: q
RHS: q,q,p
LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[q]
[q,q,p]
Checking if q is contained in
[q,q,p]
True
Sequent is Axiom!
No contradiction.
Close branch.
52. yields:
LHS: [q]
RHS: [q,q,p]

51. yields:
LHS: [q,~p]
RHS: [q,q]

50. yields:
LHS: [~q,q,~p]
RHS: [q]

right split:
Performing iteration with:
LHS: ~q,q,q
RHS: q
Non-atomic proposition found on LHS...
Iterating on the left-hand side...
54. Left- ~ on ~q with
LHS: q,q
RHS: q
Performing iteration with:
LHS: q,q
RHS: q

LHS & RHS contain only atoms!
Checking if an atom in LHS also resides in RHS:
[q,q]
[q,q]
Checking if q is contained in
[q,q]
True
Sequent is Axiom!
No contradiction.
Close branch.
54. yields:
LHS: [q,q]
RHS: [q,q]

50. yields:
LHS: [~q,q,q]
RHS: [q]

43. yields:
LHS: [(~p ∨ q),~q,q]
RHS: [q]

42. yields:
LHS: [(p ∨ q),(~p ∨ q),~q]
RHS: [q]

31. yields:
LHS: [~q,(p ∨ q),(~p ∨ q),~q]
RHS: []

30. yields:
LHS: [(p ∨ ~q),~q,(p ∨ q),(~p ∨ q)]
RHS: []

3. yields:
LHS: [((p ∨ q) ∧ (~p ∨ q)),(p ∨ ~q),~q]
RHS: []

2. yields:
LHS: [(~p ∨ ~q),((p ∨ q) ∧ (~p ∨ q)),(p ∨ ~q)]
RHS: []

1. yields:
LHS: [(((p ∨ q) ∧ (~p ∨ q)) ∧ (p ∨ ~q)),(~p ∨ ~q)]
RHS: []

0. yields:
LHS: [(((p ∨ q) ∧ (~p ∨ q)) ∧ (p ∨ ~q))]
RHS: [~(~p ∨ ~q)]