Exercise 6.6 from LEC 6 notes (page 6)

Argument:

P_1: $h \to bvt$

P_2: $b \to \neg s$

P_3: $s \wedge h \to t$

P_4: b

$\therefore C$: $\neg h$

The argument is $(P_1 \wedge P_2 \wedge P_3 \wedge P_4) \to C$

It is valid if $(P_1 \wedge P_2 \wedge P_3 \wedge P_4) \to C$ is a tautology.

To check whether a compound proposition is a tautology using a truth tree, we will instead check whether its negation is a contradiction.

Note: negation of argument $= \neg((P_1 \wedge P_2 \wedge P_3 \wedge P_4) \to C)$

$\equiv P_1 \wedge P_2 \wedge P_3 \wedge P_4 \wedge \neg C$

Thus, the root of our tree will correspond to the conjunction of all premises and the conclusion's negation.

Since conjunction is a non-splitting branching rule, we can start our tree with root

P_1: $h \to bvt$ ✓

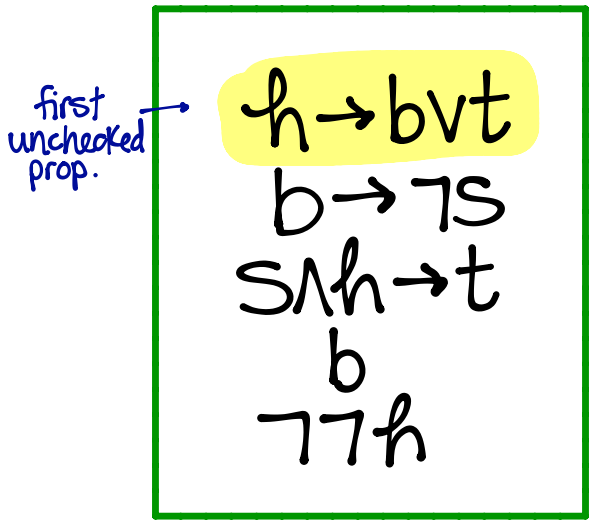
P_2: $b \to \neg s$ ✓

P_3: $s \wedge h \to t$ ✓

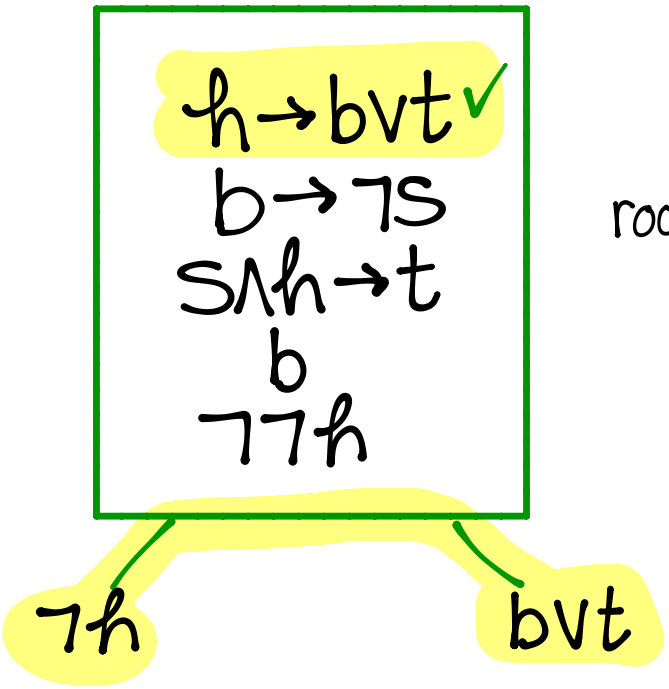
P_4: b ✓

$\neg C$: $\neg\neg h$ ✓

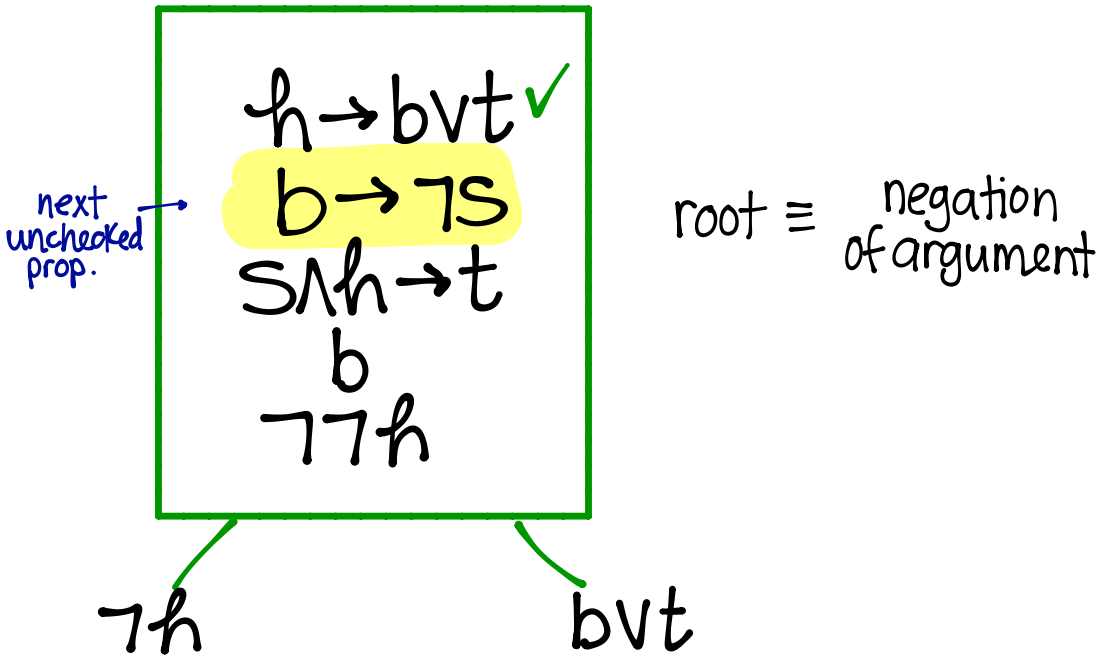
root ≡ negation of argument

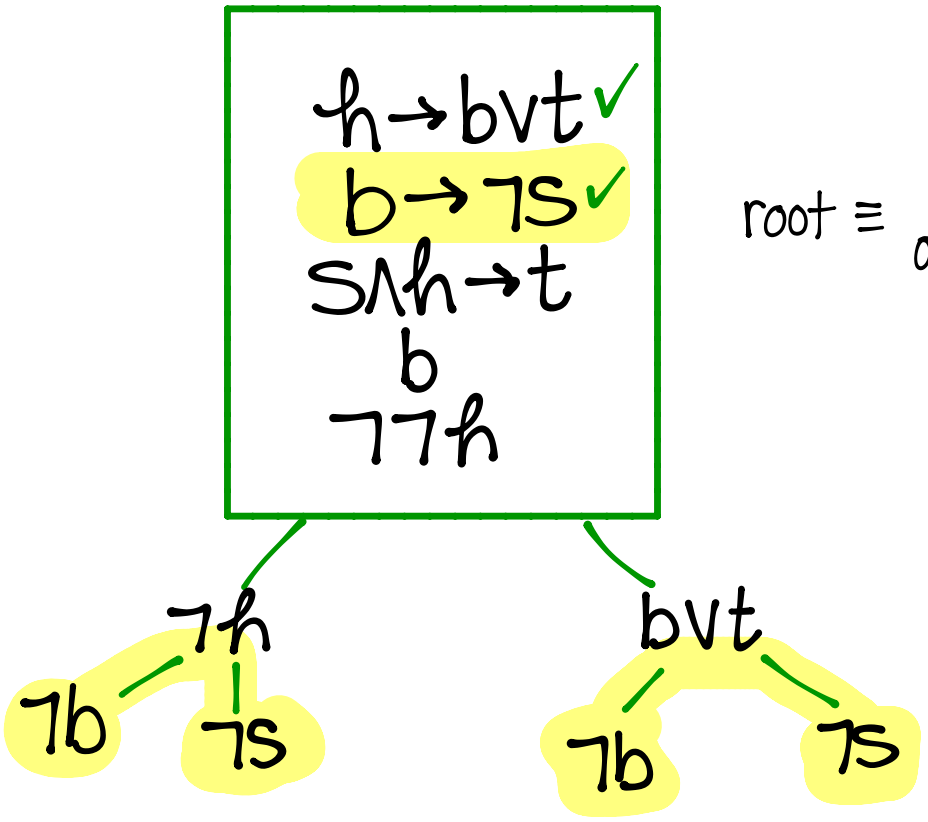


root ≡ negation of argument



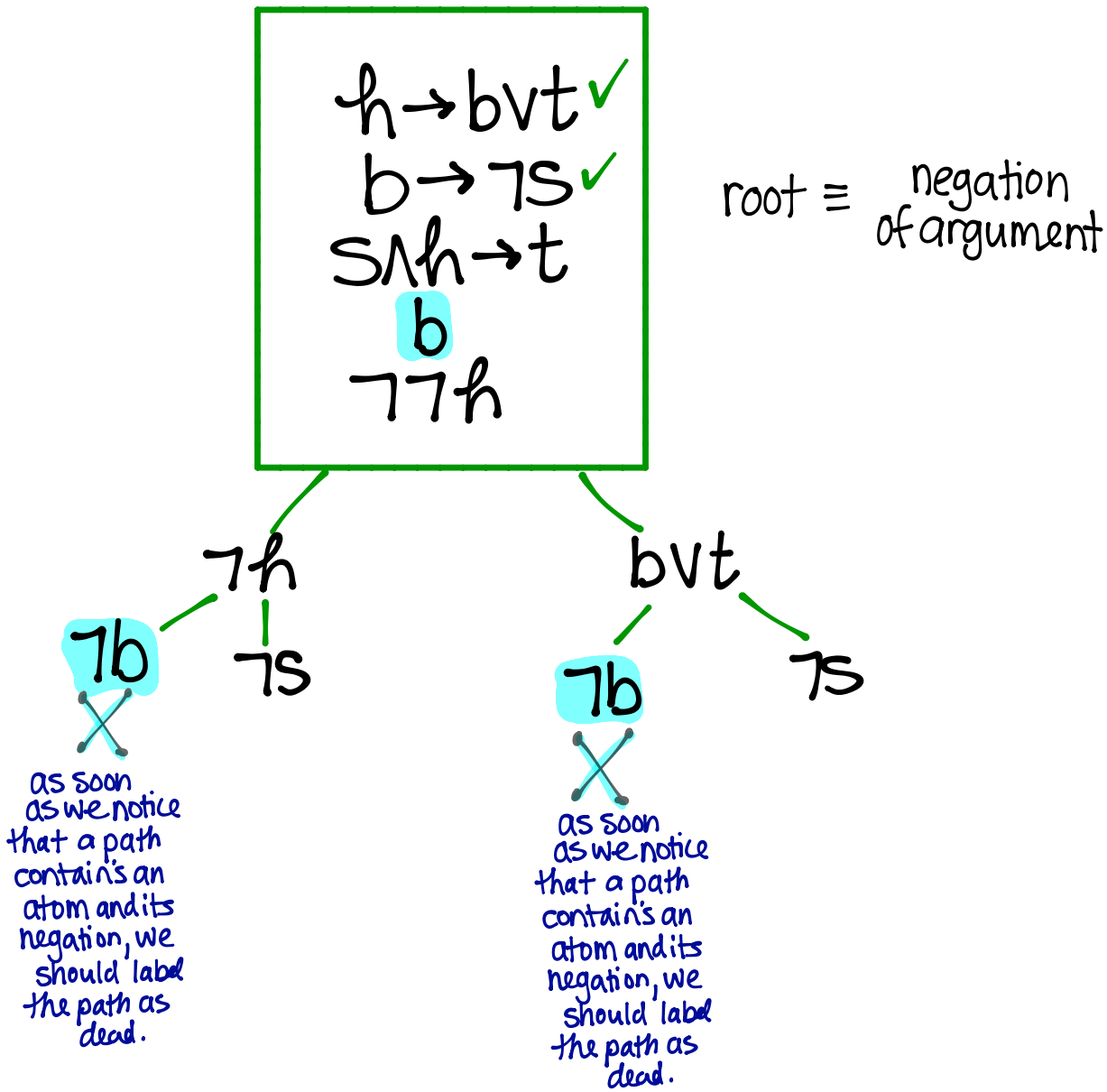
root ≡ negation of argument


h→bvt ✓
b→¬s
s∧h→t
b
¬¬h
next unchecked prop.
¬h
bvt
root ≡ negation of argument

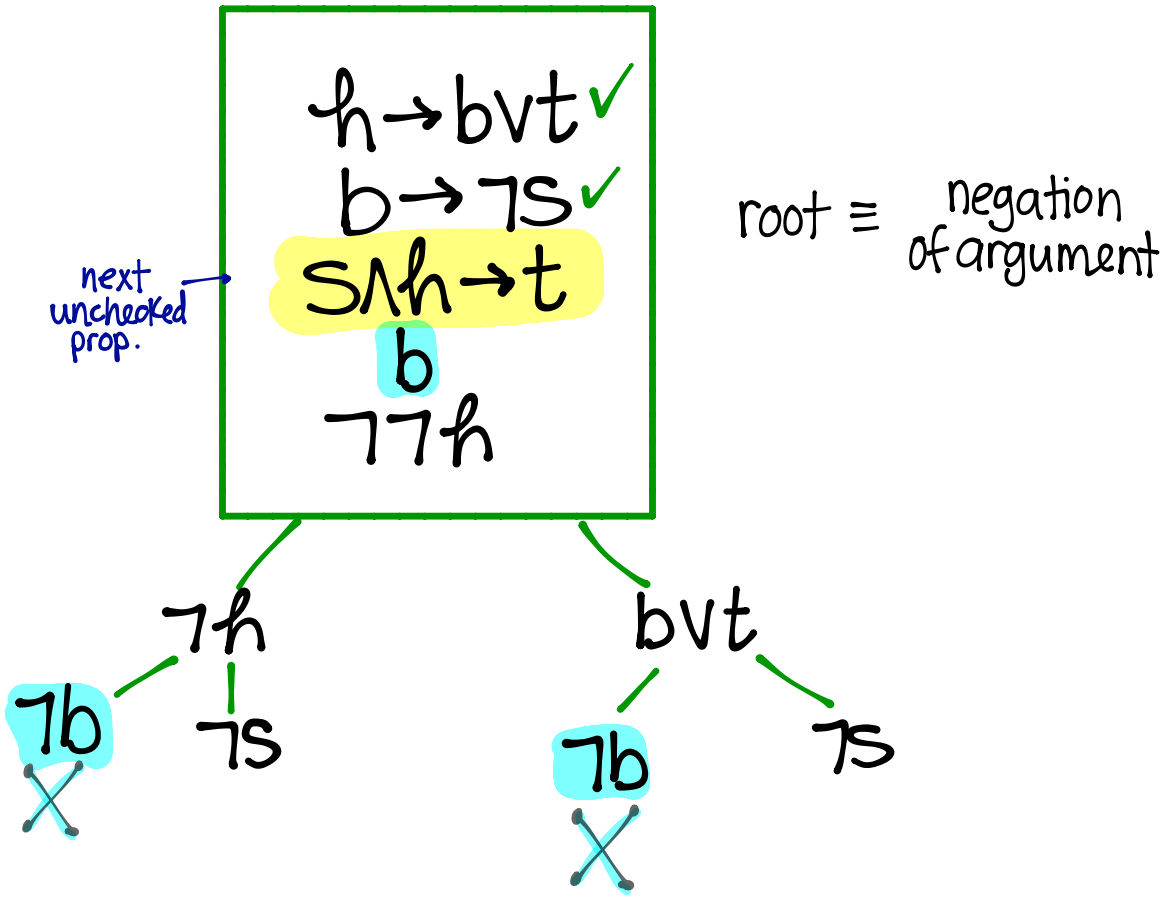


root ≡ negation of argument

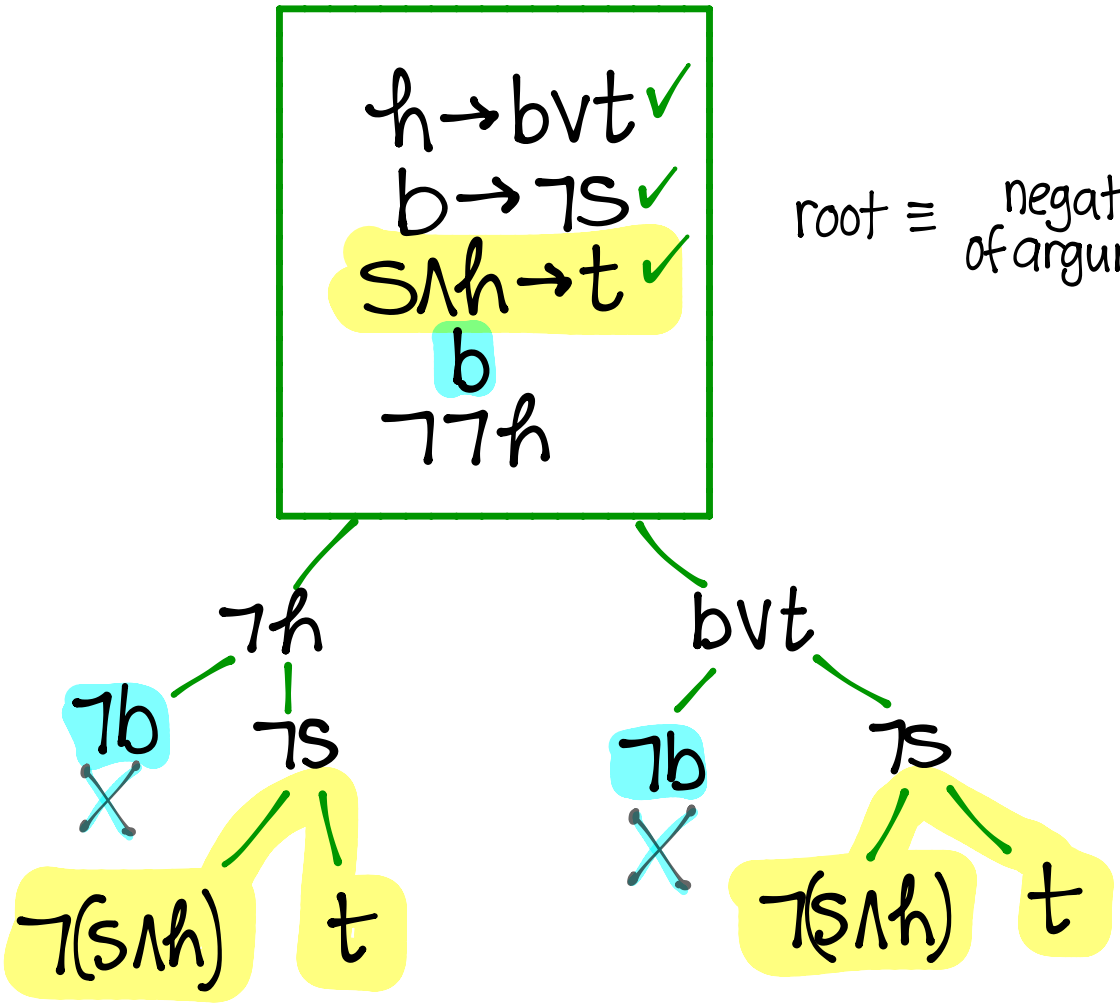
*branching rule applies to all active paths stemming down from proposition's node in tree.



root ≡ negation of argument

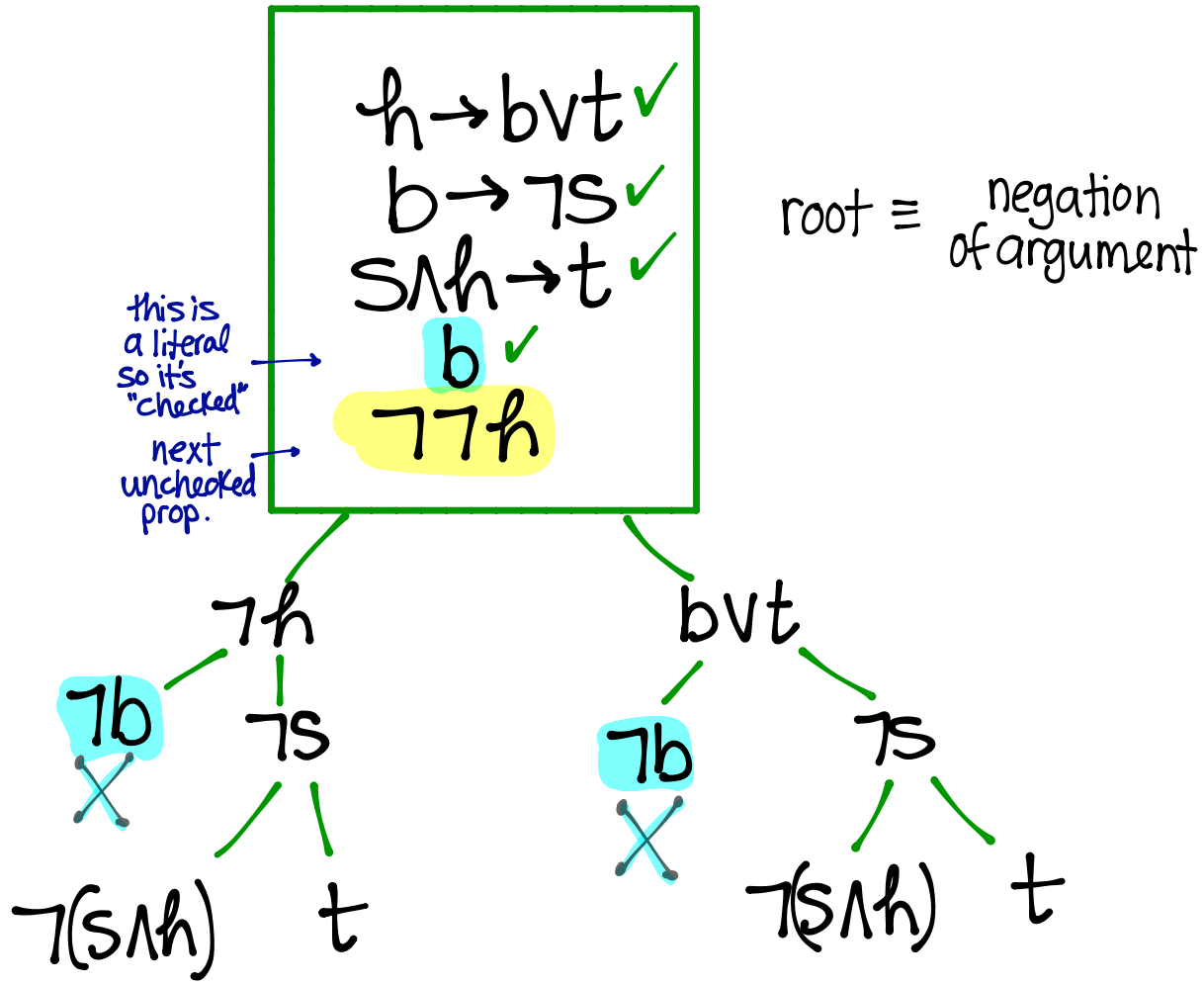


root ≡ negation of argument

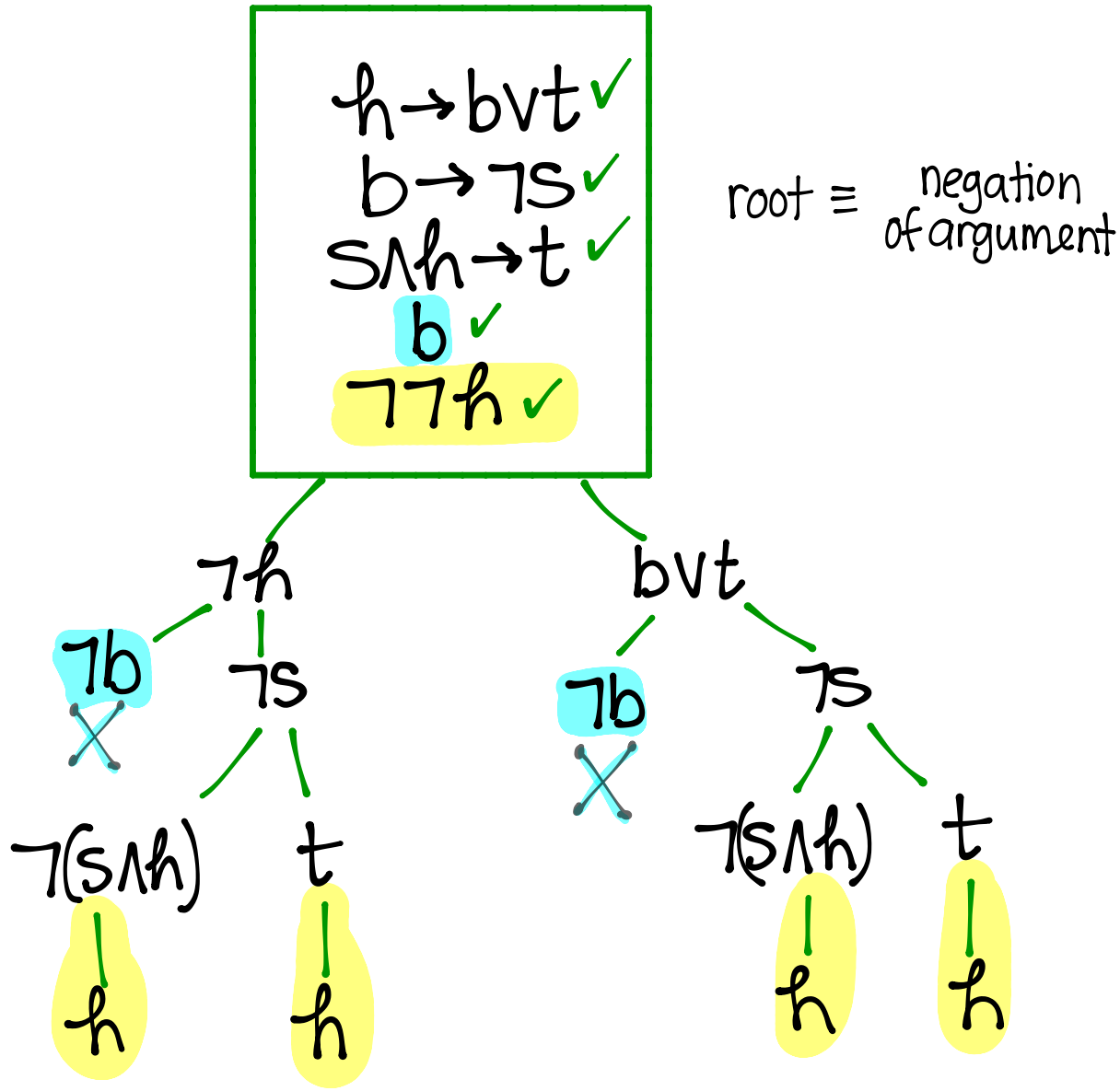


root ≡ negation of argument

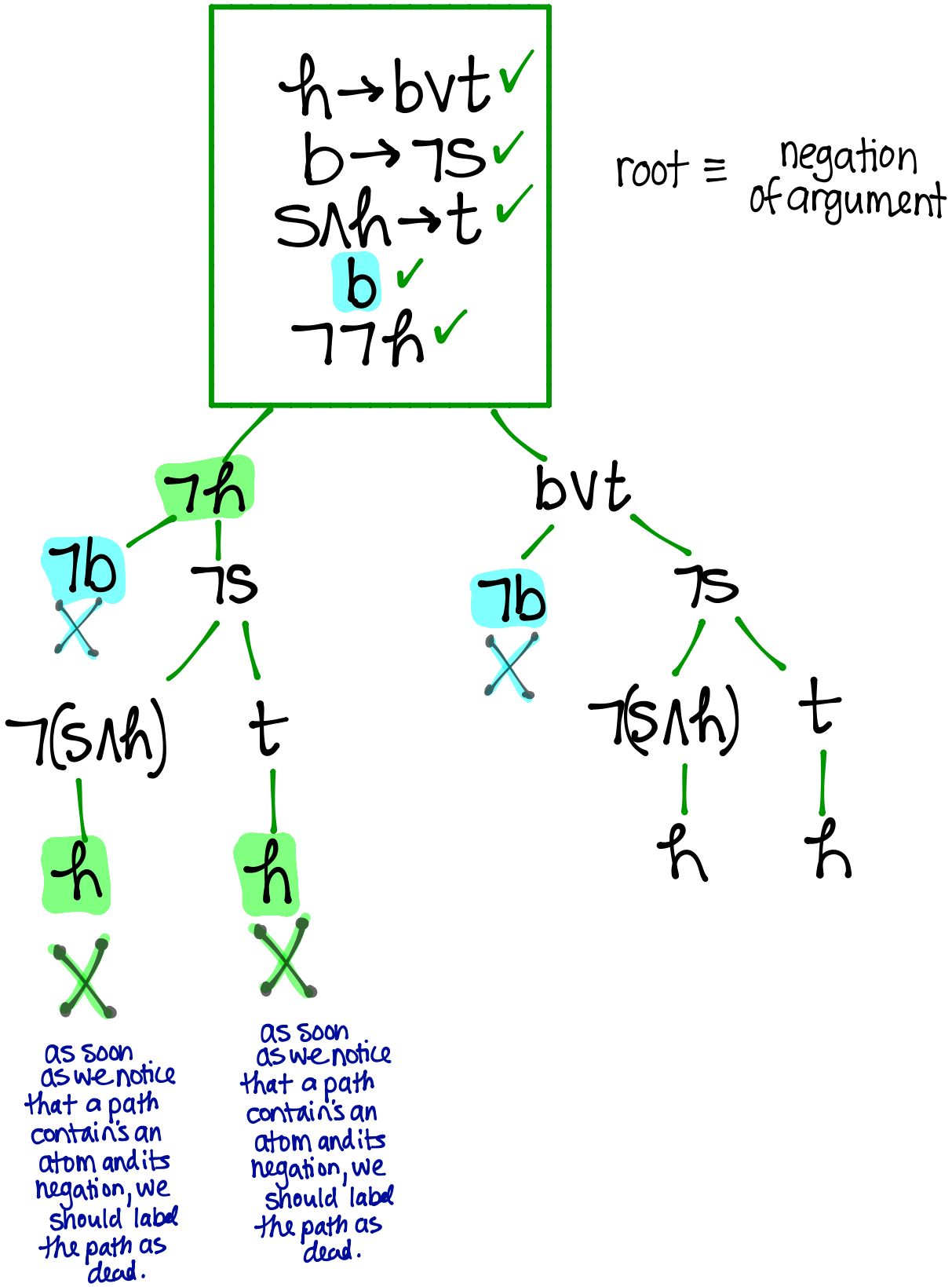
*branching rule applies to all active paths stemming down from proposition's node in tree.

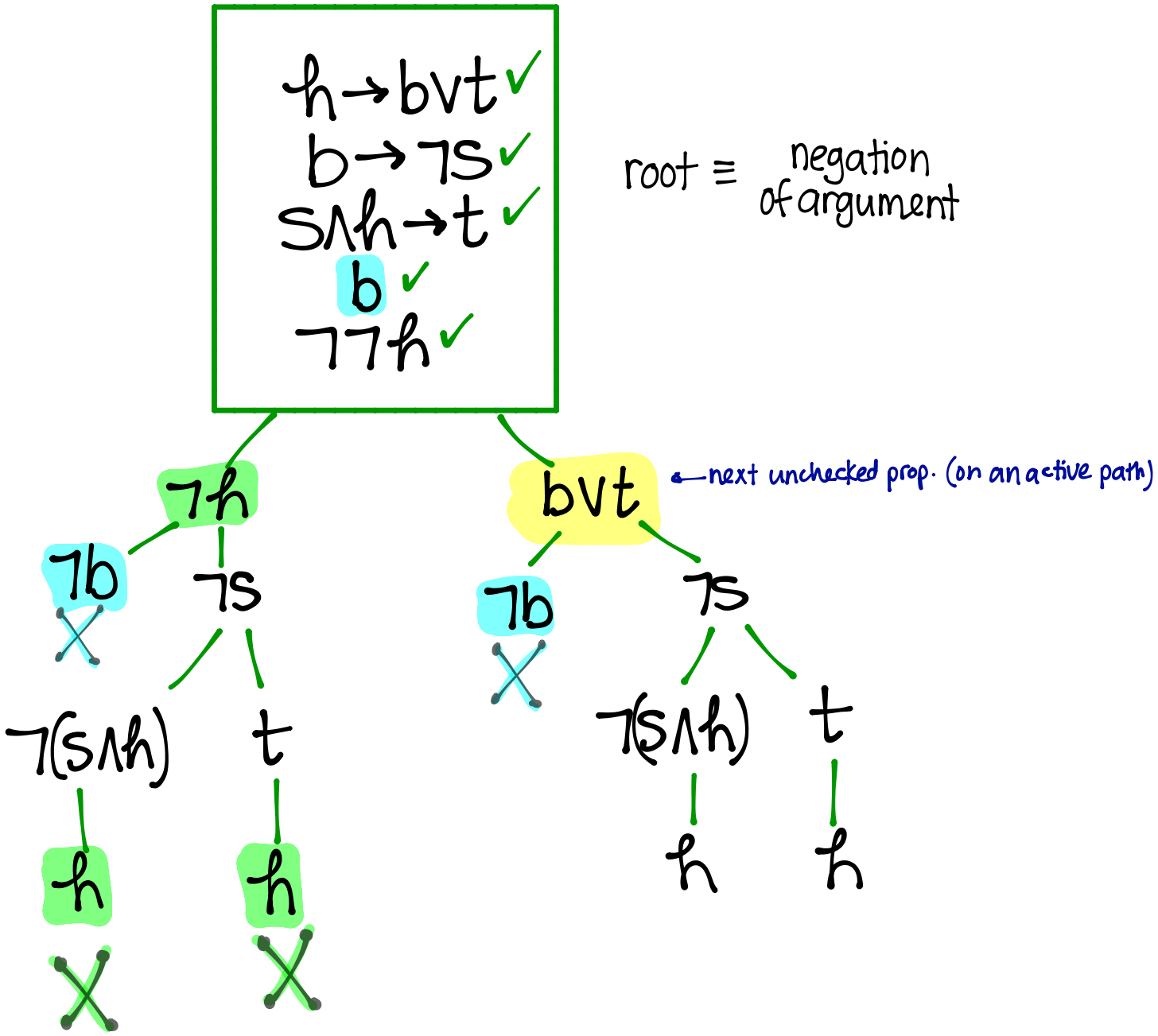


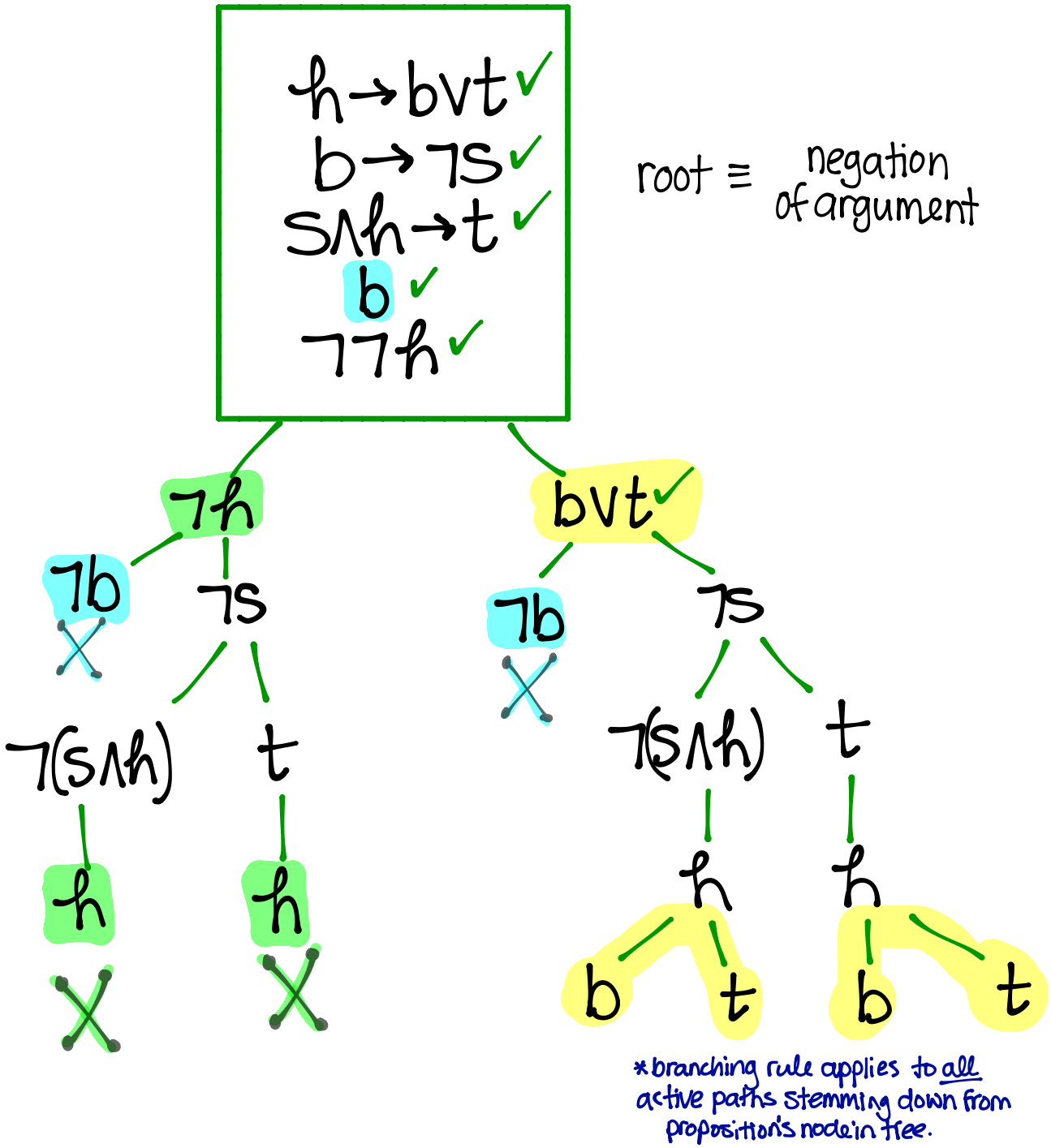
root ≡ negation of argument

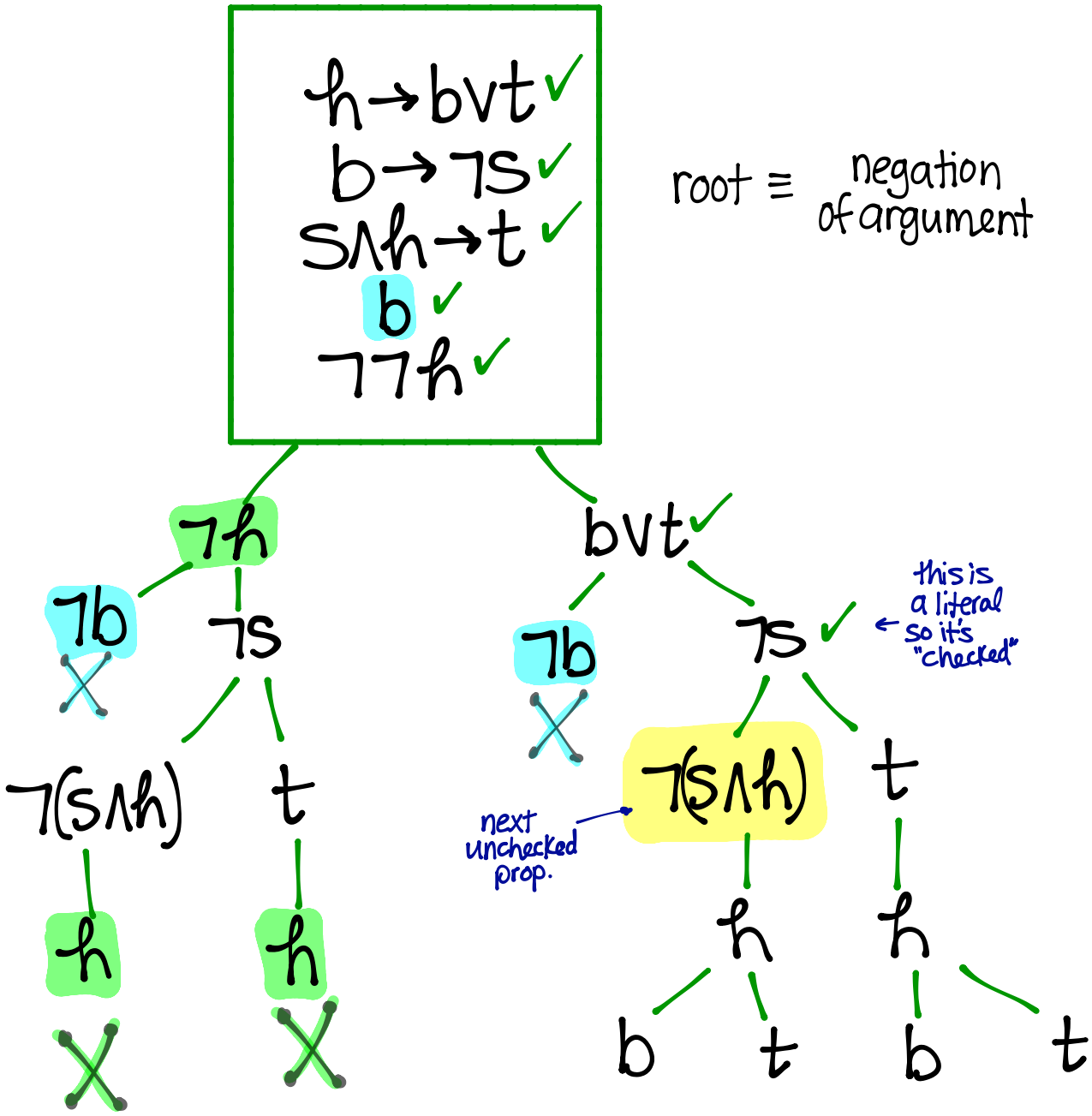


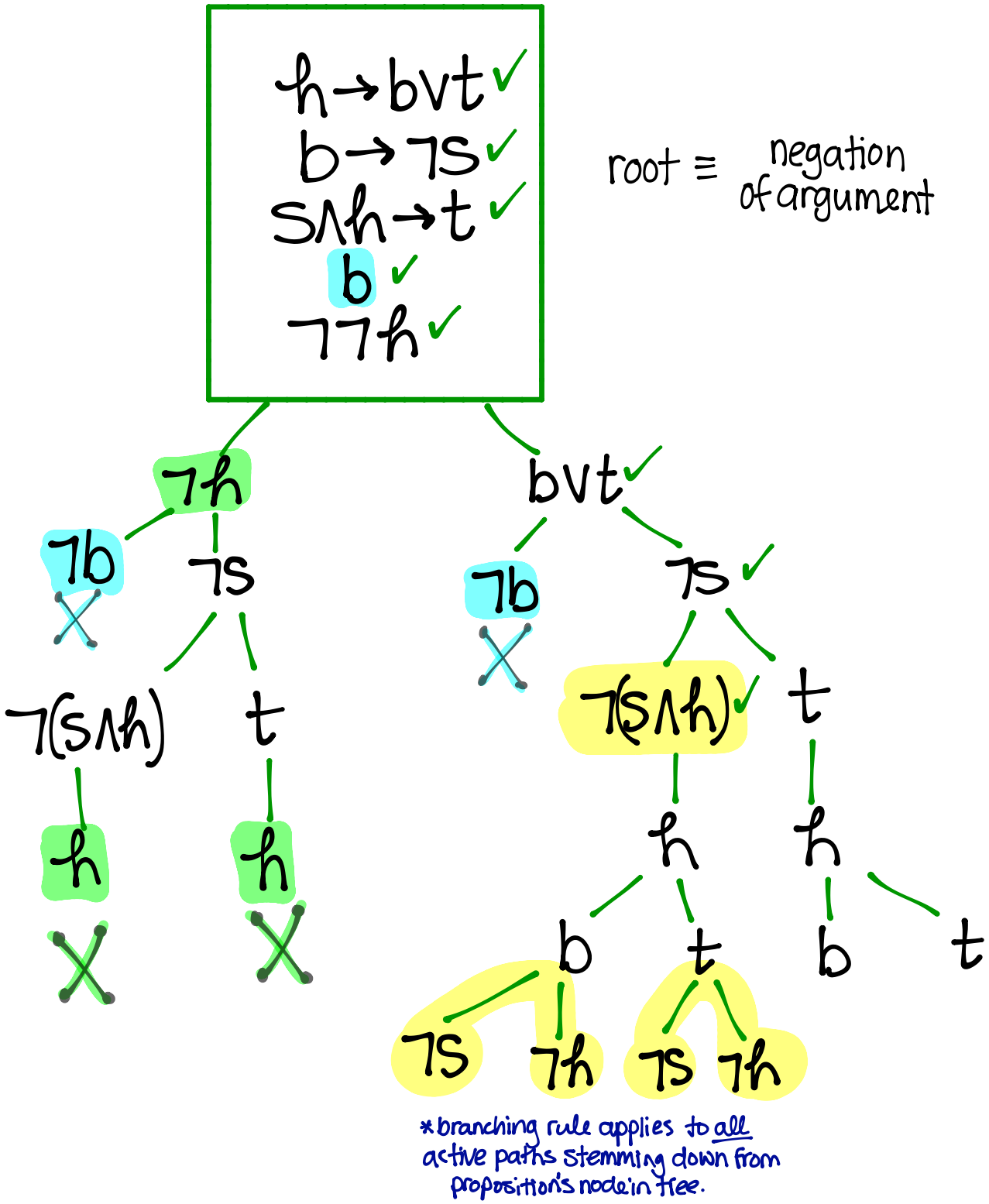
*branching rule applies to all active paths stemming down from proposition's node in tree.

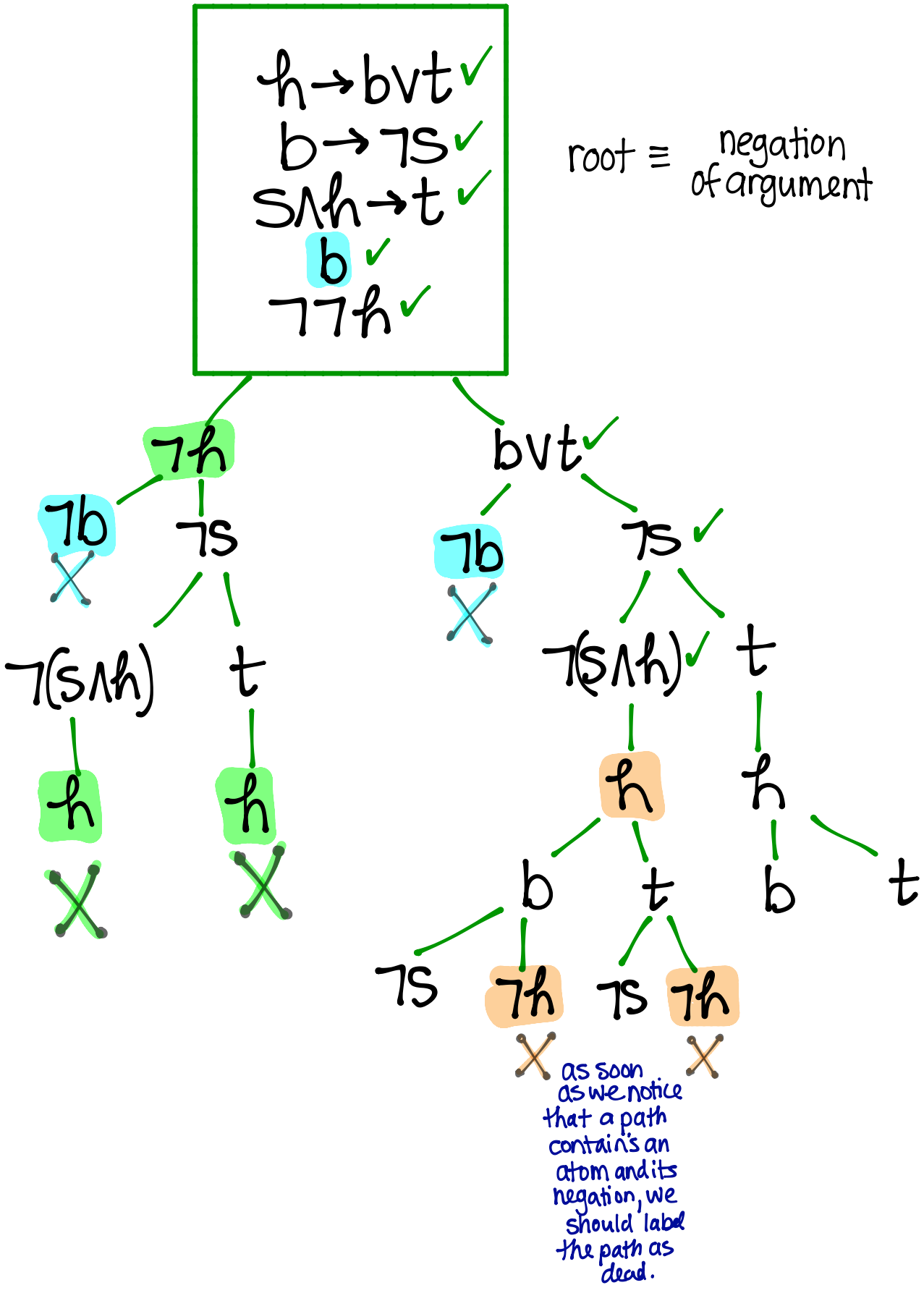

h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
bvt
¬b
¬s
¬b
¬s
¬(s∧h)
t
¬(s∧h)
t
h
h
h
h
as soon as we notice that a path contains an atom and its negation, we should label the path as dead.
as soon as we notice that a path contains an atom and its negation, we should label the path as dead.

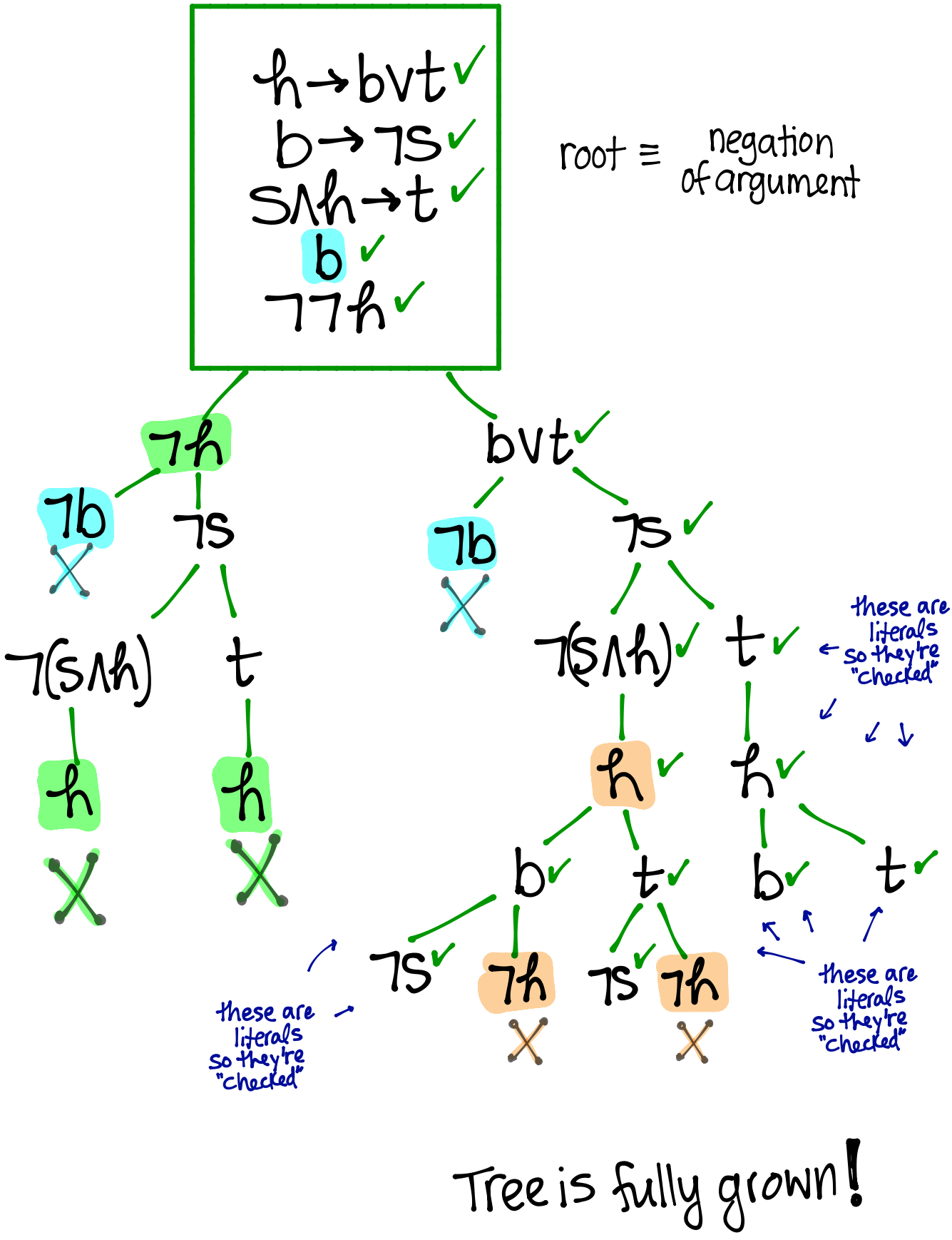

h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
bvt
← next unchecked prop. (on an active path)
¬b
¬s
¬b
¬s
¬(s∧h)
t
¬(s∧h)
t
h
h
h
h


h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
bvt ✓
¬b
¬s
¬b
¬s
¬(s∧h)
t
¬(s∧h)
t
h
h
h
h
b
t
b
t
*branching rule applies to all active paths stemming down from proposition's node in tree.

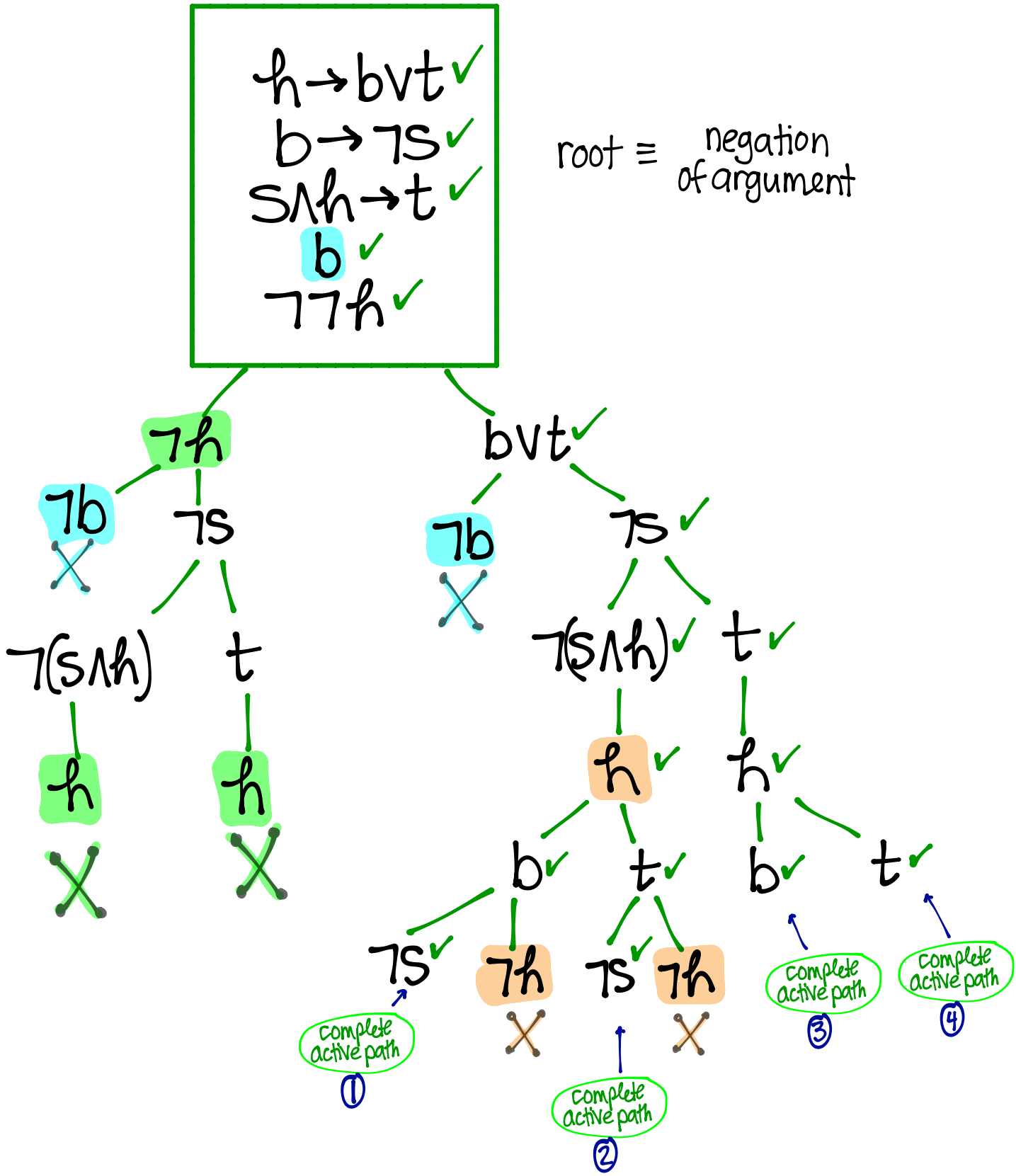

h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
X
X
bvt ✓
¬b
¬s ✓
this is a literal so it's "checked"
next unchecked prop.
¬(s∧h)
t
h
h
b
t
b
t

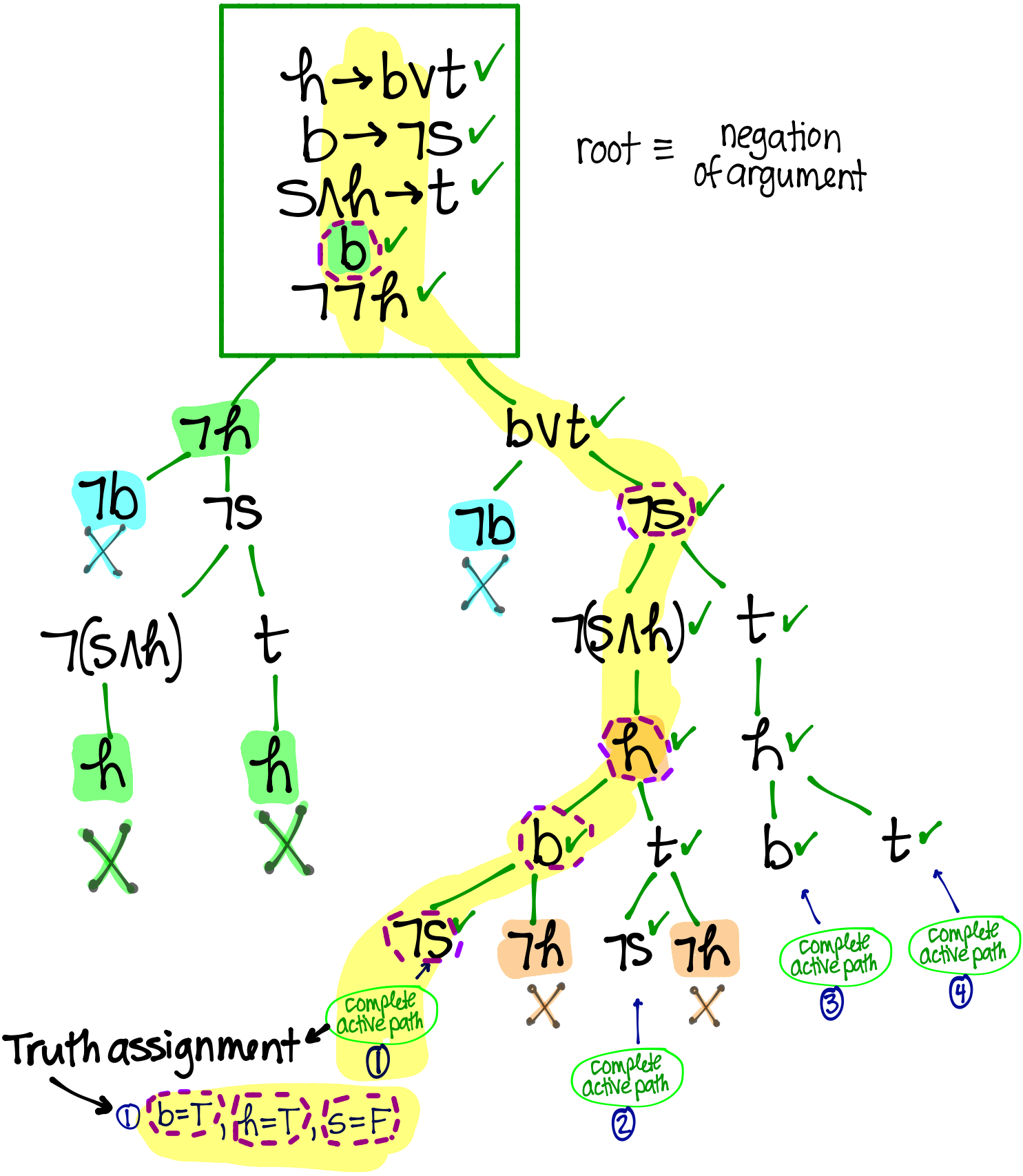

h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
X
X
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t
h
h
b
t
b
t
¬s
¬h
¬s
¬h
*branching rule applies to all active paths stemming down from proposition's node in tree.

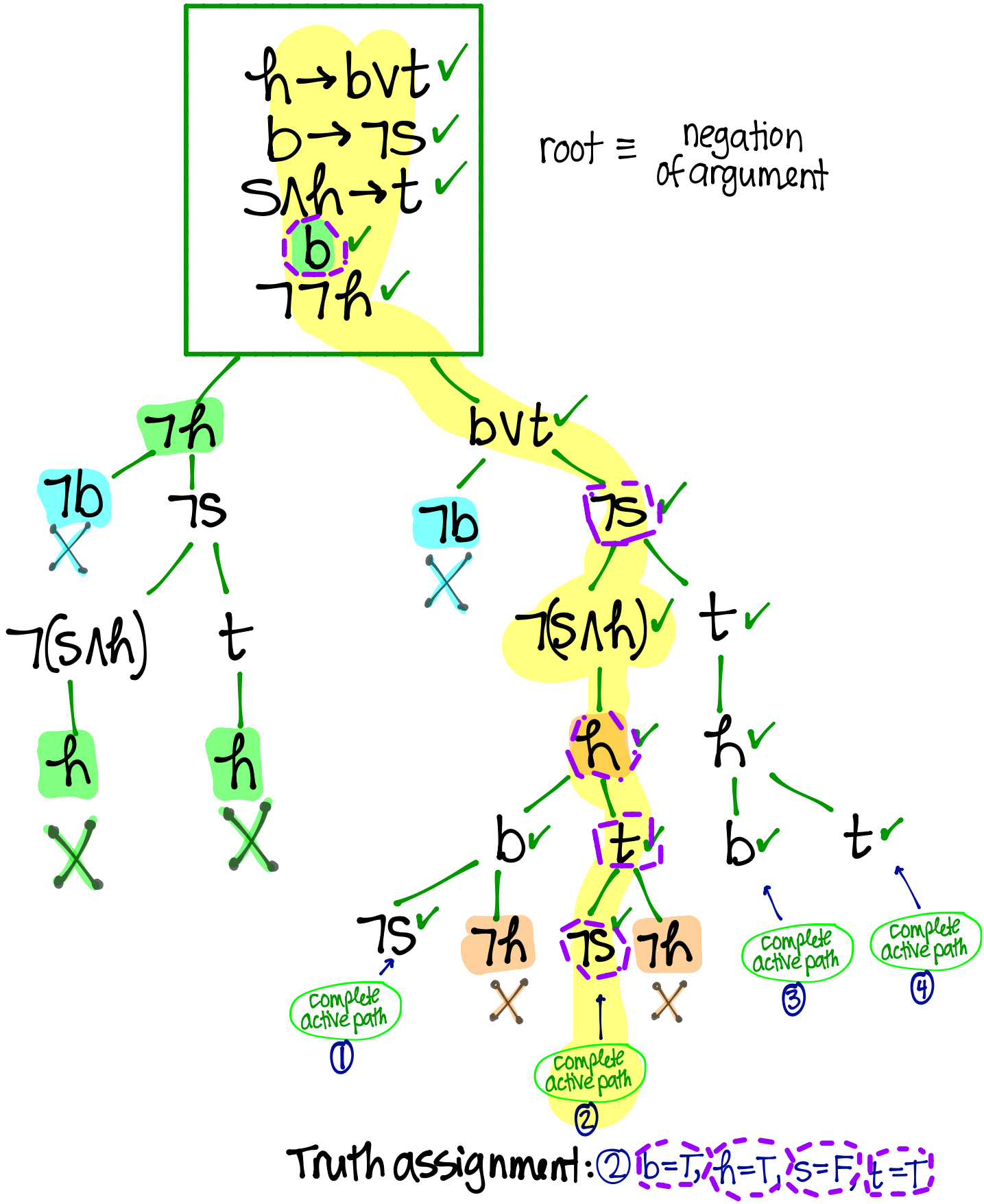

h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t
h
h
b
t
b
t
¬s
¬h
¬s
¬h
as soon as we notice that a path contains an atom and its negation, we should label the path as dead.

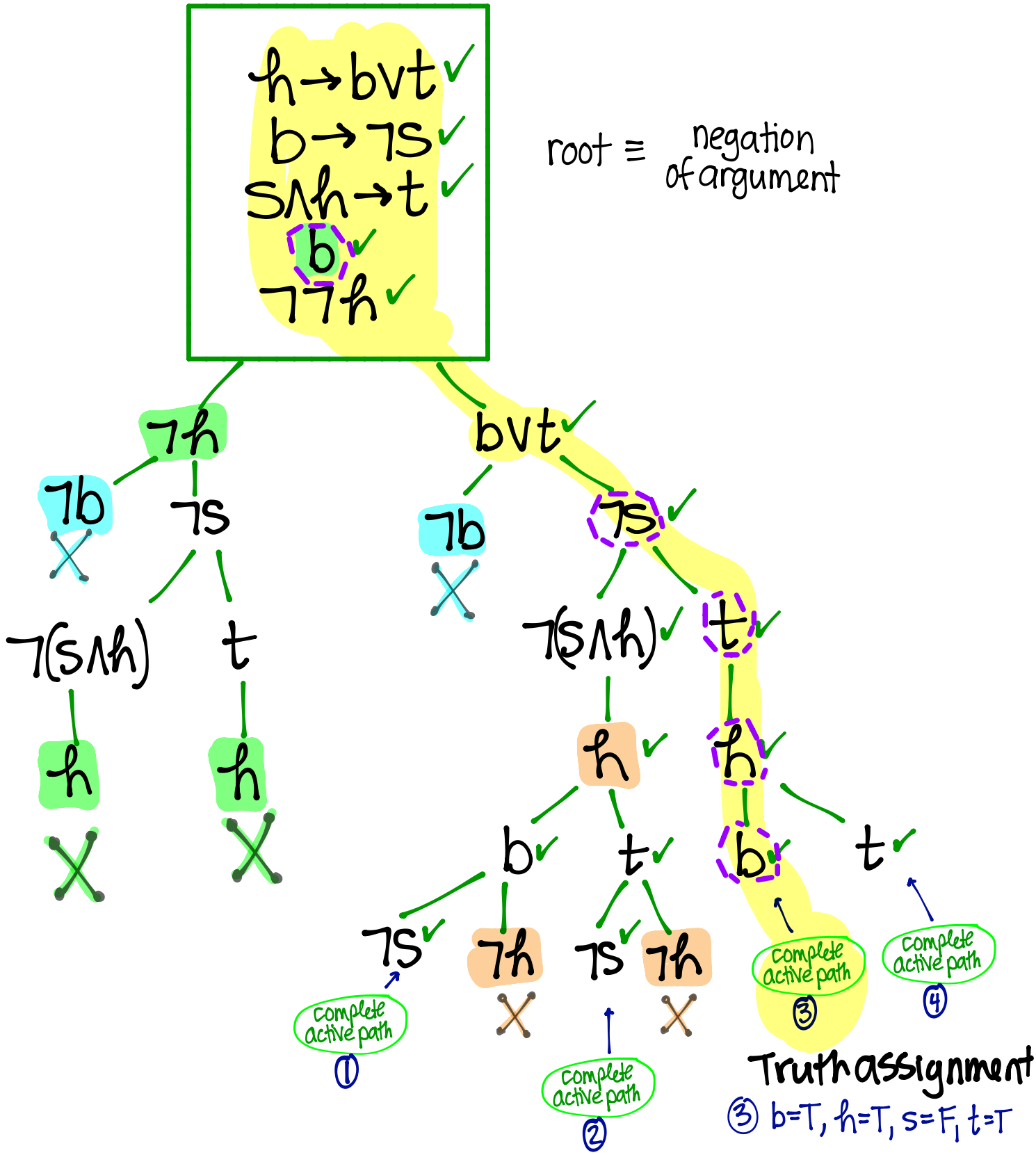


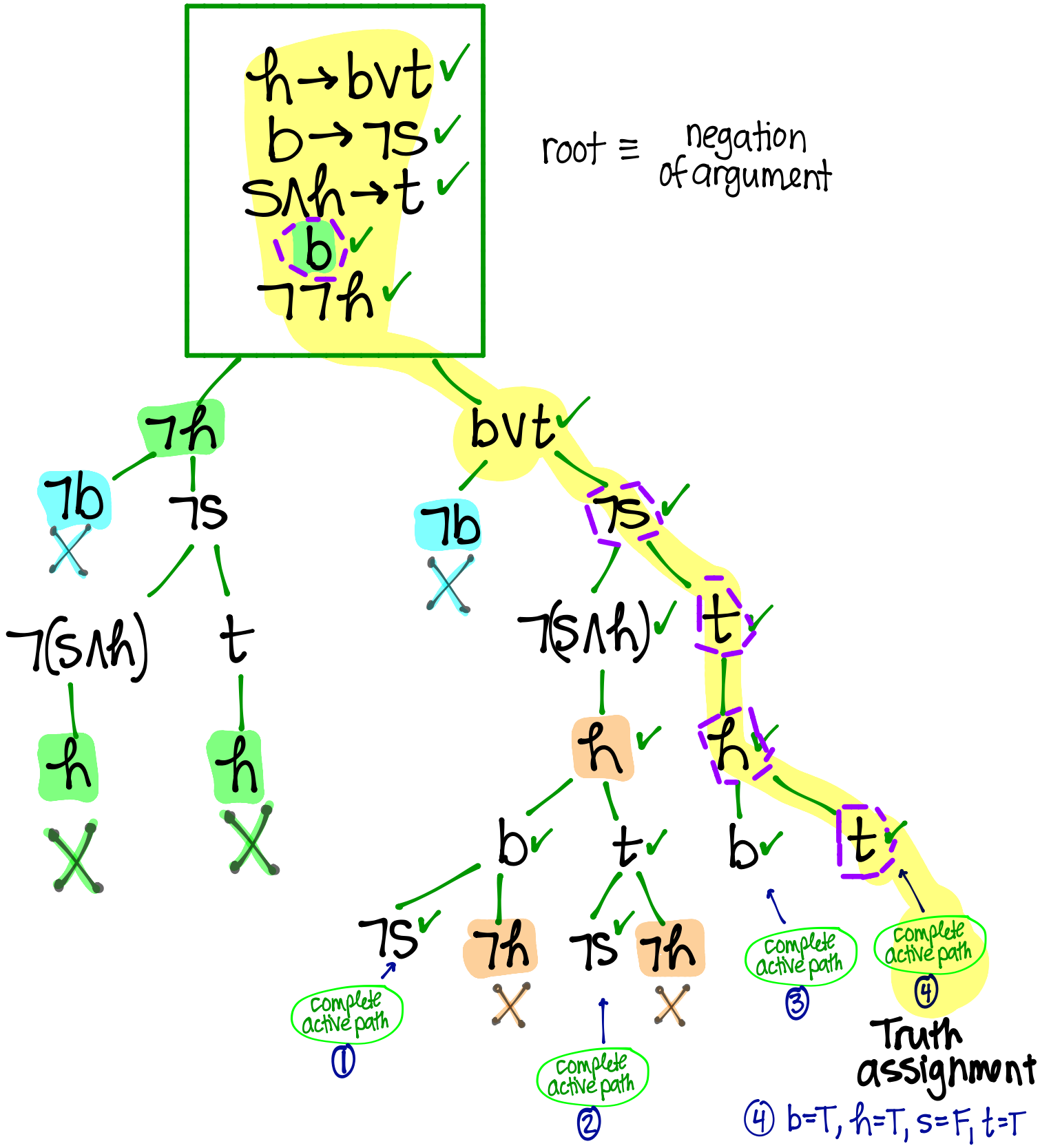
Tree is fully grown!

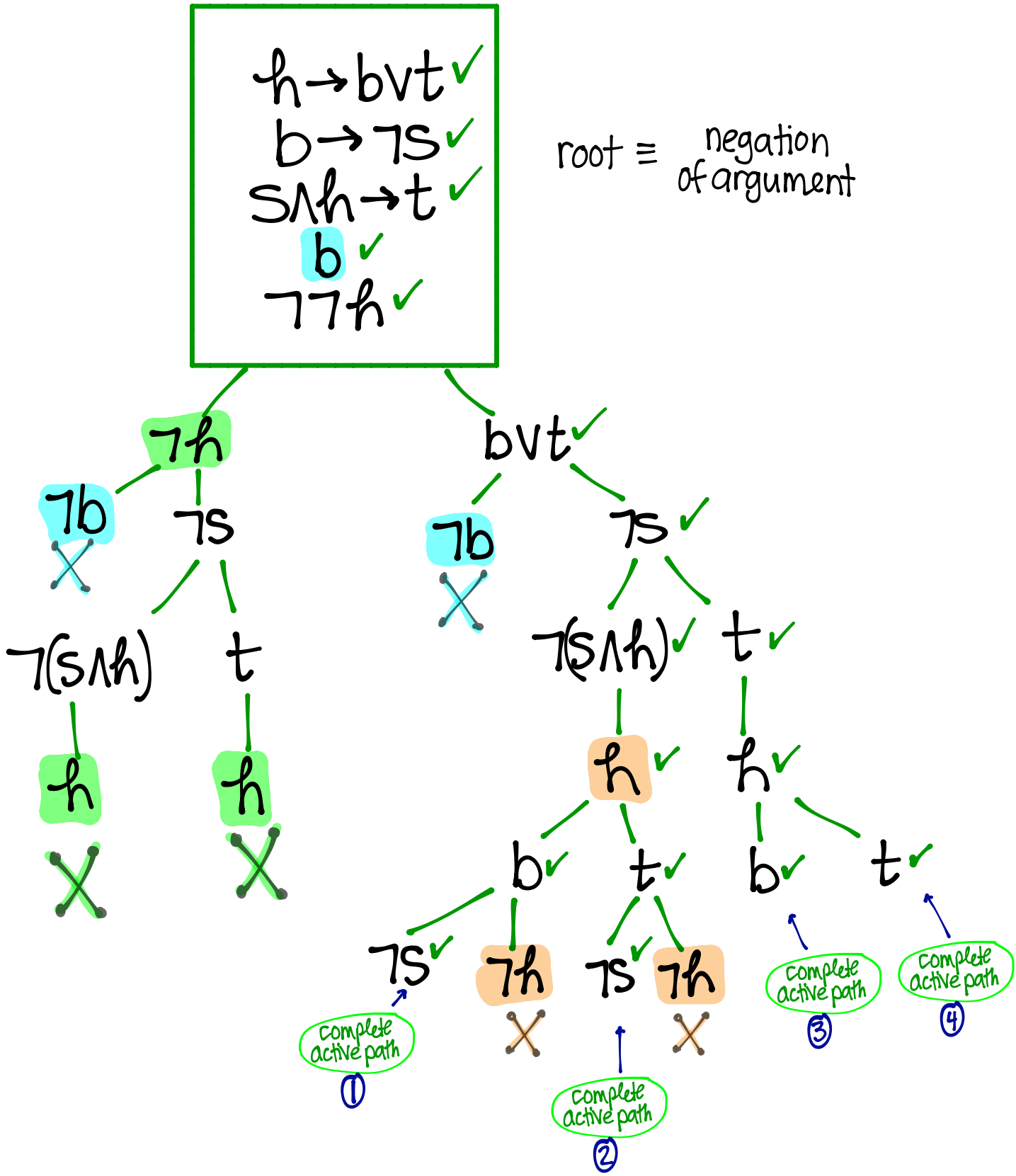

h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t ✓
h ✓
h ✓
b ✓
t ✓
b ✓
t ✓
¬s ✓
¬h
¬s ✓
¬h
complete active path ①
complete active path ②
complete active path ③
complete active path ④


h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t ✓
h ✓
h ✓
b ✓
t ✓
b ✓
t ✓
¬s ✓
¬h
¬s ✓
¬h
complete active path ①
complete active path ②
complete active path ③
complete active path ④
Truth assignment
① b=T, h=T, s=F


h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t ✓
h ✓
h ✓
b ✓
t ✓
b ✓
t ✓
¬s ✓
¬h
¬s ✓
¬h
complete active path ①
complete active path ②
complete active path ③
complete active path ④
Truth assignment: ② b=T, h=T, s=F, t=T


h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t ✓
h ✓
h ✓
b ✓
t ✓
b ✓
t ✓
¬s ✓
¬h
¬s ✓
¬h
complete active path ①
complete active path ②
complete active path ③
complete active path ④
Truth assignment
③ b=T, h=T, s=F, t=T


h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t ✓
h ✓
h ✓
b ✓
t ✓
b ✓
t ✓
¬s ✓
¬h
¬s ✓
¬h
complete active path ①
complete active path ②
complete active path ③
complete active path ④
Truth assignment
④ b=T, h=T, s=F, t=T


h→bvt ✓
b→¬s ✓
s∧h→t ✓
b ✓
¬¬h ✓
root ≡ negation of argument
¬h
¬b
¬s
¬(s∧h)
t
h
h
bvt ✓
¬b
¬s ✓
¬(s∧h) ✓
t ✓
h ✓
h ✓
b ✓
t ✓
b ✓
t ✓
¬s ✓
¬h
¬s ✓
¬h
complete active path ①
complete active path ②
complete active path ③
complete active path ④