Completeness Theorem:

Fact 1: A |- α iff A |= α. If A = ∅, then **|= α** means that α is a tautology.

Propositional formula α is indecomposable iff α is a literal or negation of a literal

Example: A, ~A

Sequence of formulas is **fundamental** if it contains a formula and its negation.

Example 1: A, A->(BvC), A->C, BvA, ~(A->C), CvA.

RS -strategy converts a formula α to a decomposition tree T(α) using 7 decomposition rules.

Disjunction decomposition rules

$$(\cup)\ \frac{\Gamma', (A \cup B), \Delta}{\Gamma', A, B, \Delta}, \qquad (\neg\cup)\ \frac{\Gamma', \neg(A \cup B), \Delta}{\Gamma', \neg A, \Delta \quad : \quad \Gamma', \neg B, \Delta}$$

Conjunction decomposition rules

$$(\cap)\ \frac{\Gamma', (A \cap B), \Delta}{\Gamma', A, \Delta \quad ; \quad \Gamma', B, \Delta}, \qquad (\neg\cap)\ \frac{\Gamma', \neg(A \cap B), \Delta}{\Gamma', \neg A, \neg B, \Delta}$$

Implication decomposition rules

$$(\Rightarrow)\ \frac{\Gamma', (A \Rightarrow B), \Delta}{\Gamma', \neg A, B, \Delta}, \qquad (\neg \Rightarrow)\ \frac{\Gamma', \neg(A \Rightarrow B), \Delta}{\Gamma', A, \Delta \quad : \quad \Gamma', \neg B, \Delta}$$

Negation decomposition rule

$$(\neg\neg)\ \frac{\Gamma', \neg\neg A, \Delta}{\Gamma', A, \Delta}$$

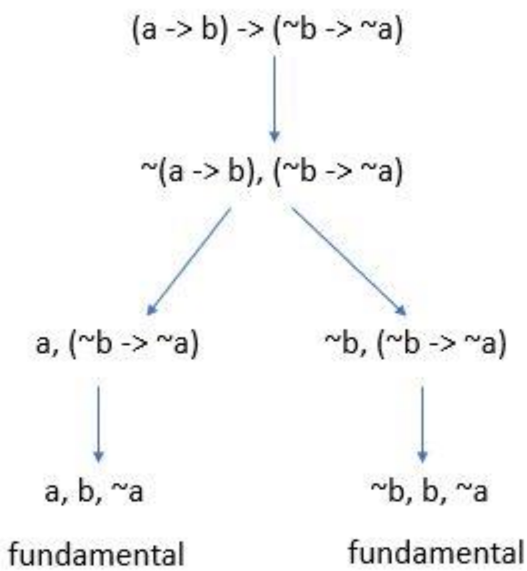
where Γ’, Δ are sequences of formulas and Γ’ is indecomposable (all formulas in Γ’ are indecomposable)

Decomposition tree **T(α), for a formula α is built from left to right**!!!!

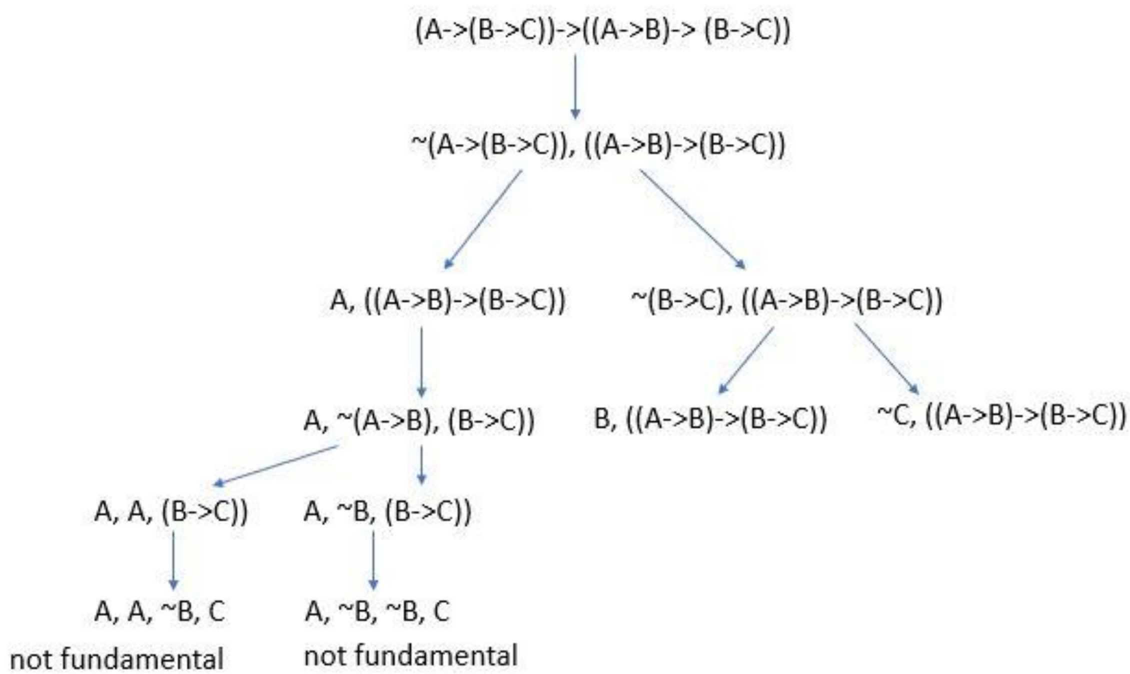
Theorem:

|= α iff every leaf in T(α) is fundamental.

Example 2: Decomposition tree **T**((a->b) ->(~b -> ~a)).



Example 3: Decomposition tree **T**((A->(B->C))-> ((A->B)->(B->C))).



Example 4:

Build decomposition tree for (A->(B->C))->((A->B)-> (A->C)) using RS decomposition rules.

Remark: Use notation ";" when a node splits to two nodes.

(A->(B->C))->((A->B)-> (A->C))

~(A->(B->C)), ((A->B)-> (A->C))

A, ((A->B)-> (A->C)); ~(B->C), ((A->B)-> (A->C))

A, ~(A->B), (A->C); ~(B->C), ((A->B)-> (A->C))

A, A, (A->C); A, ~B, (A->C); ~(B->C), ((A->B)-> (A->C))

A, A, ~A, C; A, ~B, ~A, C; ~(B->C), ((A->B)-> (A->C))

fundamental; fundamental;

So, we continue with

~(B->C), ((A->B)-> (A->C))

B, ((A->B)-> (A->C)); ~C, ((A->B)-> (A->C))

B, ~(A->B), (A->C) ; ~C, ((A->B)-> (A->C))

B, A, (A->C) ; B, ~B, (A->C) ; ~C, ((A->B)-> (A->C))

B, A, ~A, C ; B, ~B, (A->C) ; ~C, ((A->B)-> (A->C))

fundamental; fundamental;

So, we continue with

~C, ((A->B)-> (A->C))

~C, ~(A->B), (A->C)

~C, A, (A->C); ~C, ~B, (A->C)

~C, A, ~A, C; ~C, ~B, ~A, C

fundamental; fundamental

So, all leaves of the decomposition tree are fundamental which means

(A->(B->C))->((A->B)-> (A->C)) is a tautology.