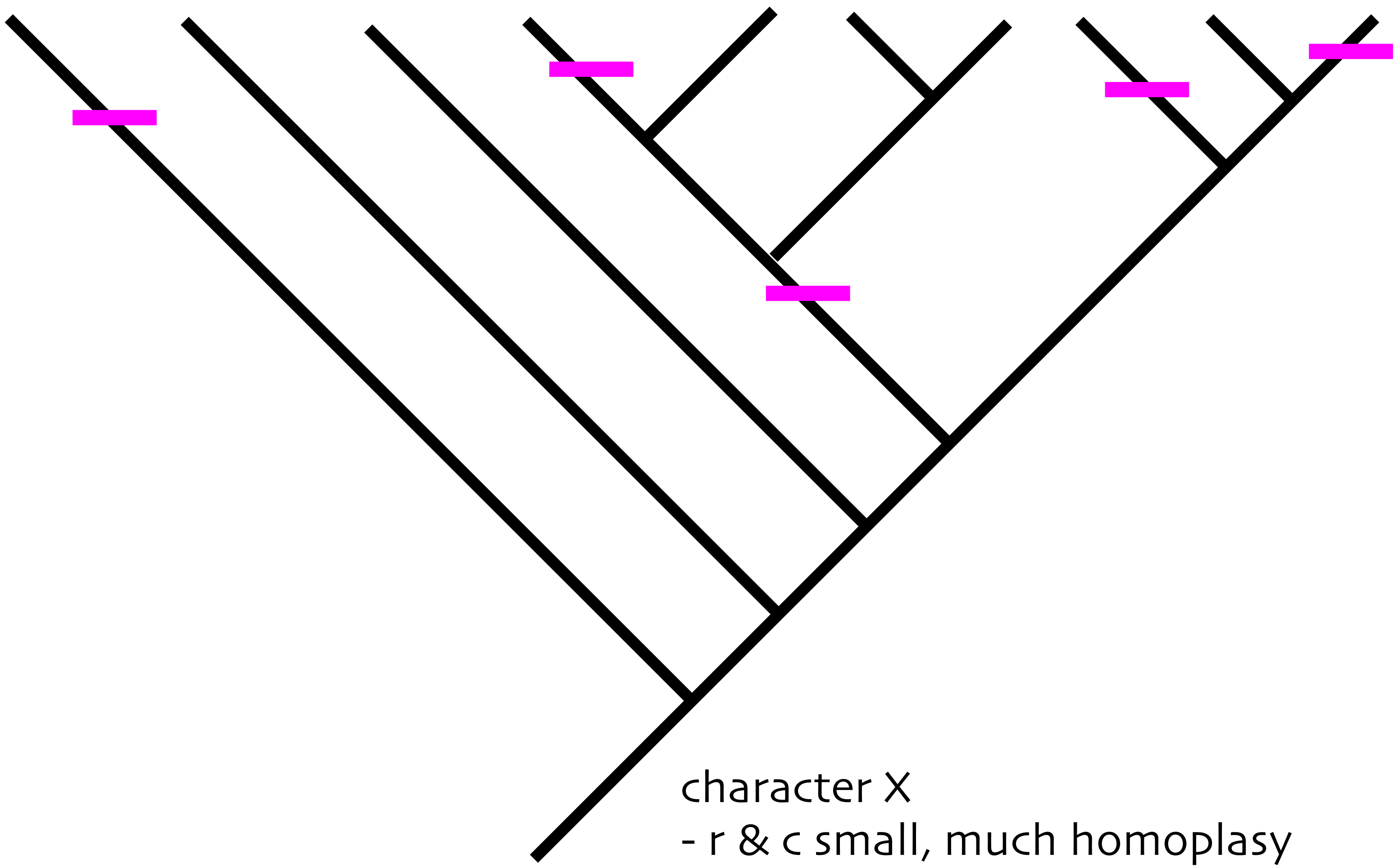
13.xi.

1. *a posteriori* weighting

2. optimization

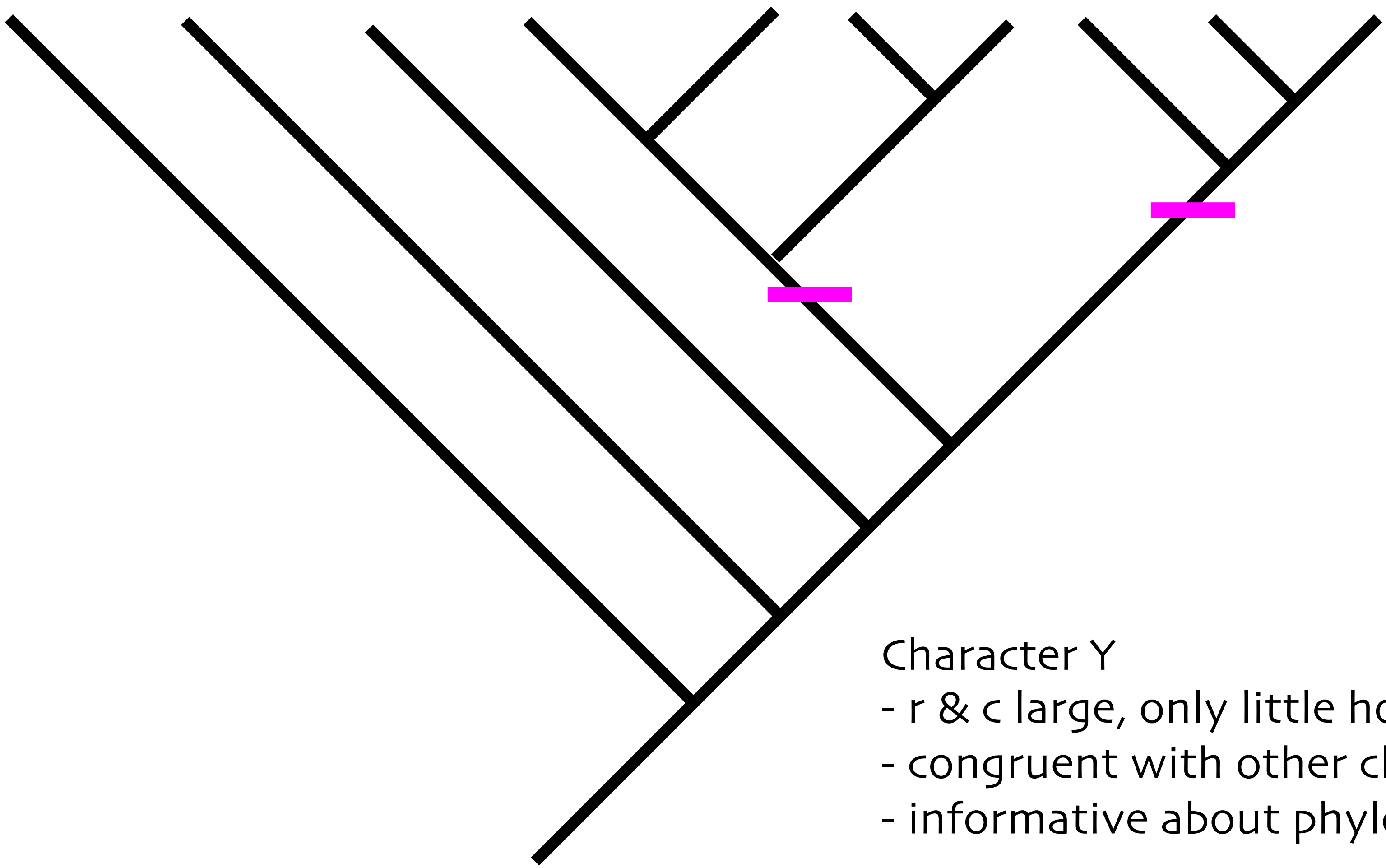
3. summary

CHARACTER WEIGHT



character X
- r & c small, much homoplasy
- in conflict with other characters
- poor information about phylogeny

CHARACTER WEIGHT



Character Y
- r & c large, only little homoplasy
- congruent with other characters
- informative about phylogeny

CHARACTER WEIGHT

information from ch. Y is more reliable than from ch. X

in analysis more weight for Y than X

weighting made using rescaled consistency index (r x c)

CHARACTER WEIGHT

A POSTERIORI (after analysis)

1) cladistic analysis

2) r x c calculated for all characters on shortest tree

3) characters weighted with the value of r x c

4) re-analysis

5) back to 2)

> Farris, J.S. 1969. A successive approximations approach to character weighting. *Systematic Zoology* 18:374-385.

2-5 repeated until result stabilizes (tree length & ch. weights), i.e. in two analyses following each other same result obtained

CHARACTER WEIGHT

central assumption in *a posteriori* weighting is that characters with lots of homoplasy (those WITHOUT reliable signal about evolutionary history) are not as reliable hypotheses of homology as characters with very little homoplasy (part of historical signal), i.e. congruent with other characters

IMPLIED CHARACTER WEIGHTING

Goloboff, P.A. 1993. Estimating character weights during tree search. *Cladistics* 9:83-91.

differential weighting is performed from the very start of the analysis

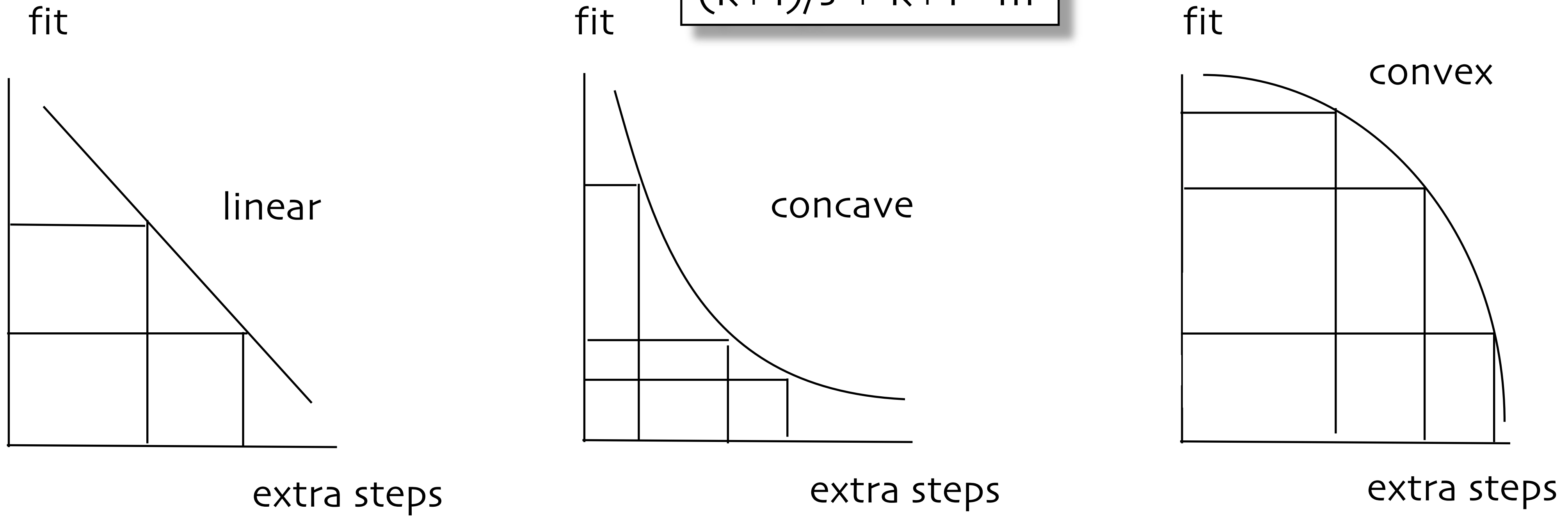
weighting performed during analysis using consistency related index

c = m/s, CONCAVE weighting function

CHARACTER WEIGHT

concavity might be TOO severe, reduced by using constant k (e.g. 3-20)

(k+1)/s + k+1 - m



IMPLIED CHARACTER WEIGHTING

differential weighting is performed from the very start of the analysis

weighting performed during analysis using consistency related index $(k+1)/s + k+1 - m$

this means that increase/decrease of ch. state changes in characters with less homoplasy (high index value) affect result more than same kind of changes in characters with much homoplasy (low index value) ---->

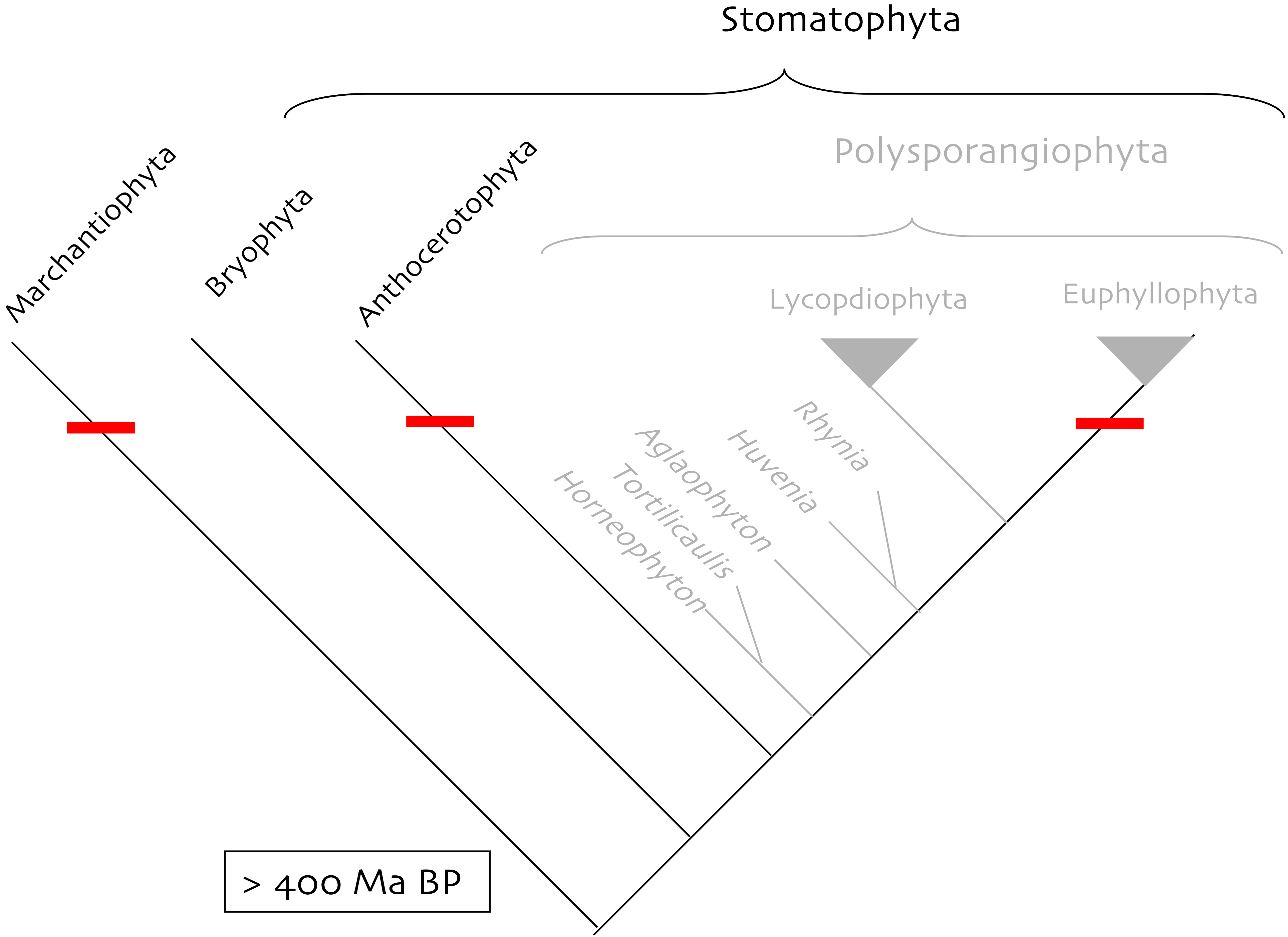
IMPLIED CHARACTER WEIGHTING

differential weighting is performed from the very start of the analysis

weighting performed during analysis using consistency related index (k+1)/s + k+1 - m

this means that increase/decrease of ch. state changes in characters with less homoplasy affect result more than same kind of changes in characters with much homoplasy (low index value) ----> preference of trees were CHARACTERS WITH LESS HOMOPLASY ARE MORE DECISIVE

instead of trying to find a tree with smallest number of ch. state changes this approach tries to find a TREE MAXIMIZING FIT


Stomatophyta
Polysporangiophyta
Marchantiophyta
Bryophyta
Anthocerotophyta
Lycopdiophyta
Euphyllophyta
Rhynia
Huvenia
Aglaophyton
Tortilicaulis
Horneophyton
> 400 Ma BP

Källersjö, M., V.A. Albert, & J.S. Farris 1999. Homoplasy increases phylogenetic structure. *Cladistics* 15:91-95

nucleotide triplets coding for aminoacids

triplets for **Pro**lin

CCT

CCC

CCA

CCG

synonymous vs. nonsynonymous substitutions

INDICES DESCRIBING TREES

individual character

ALL characters (ensemble indices

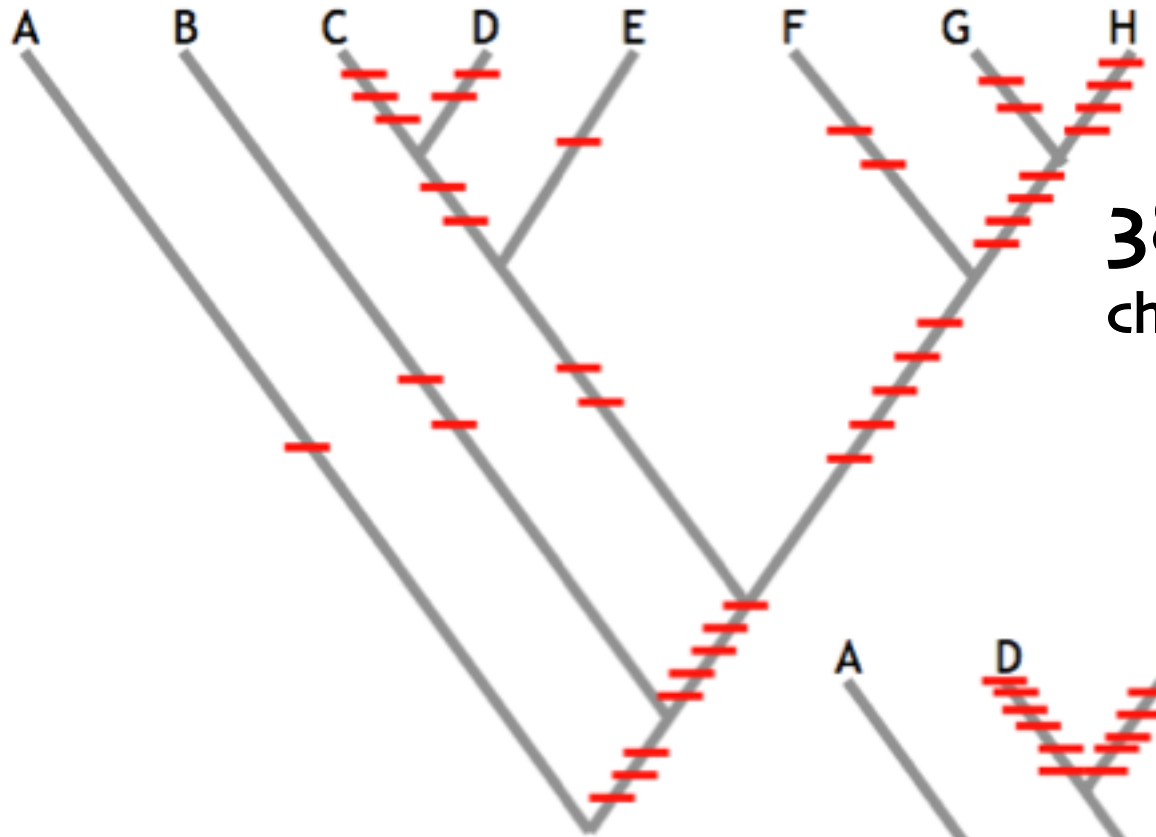
$$c = \frac{m}{s}$$

$$C = \frac{\sum m}{\sum s}$$

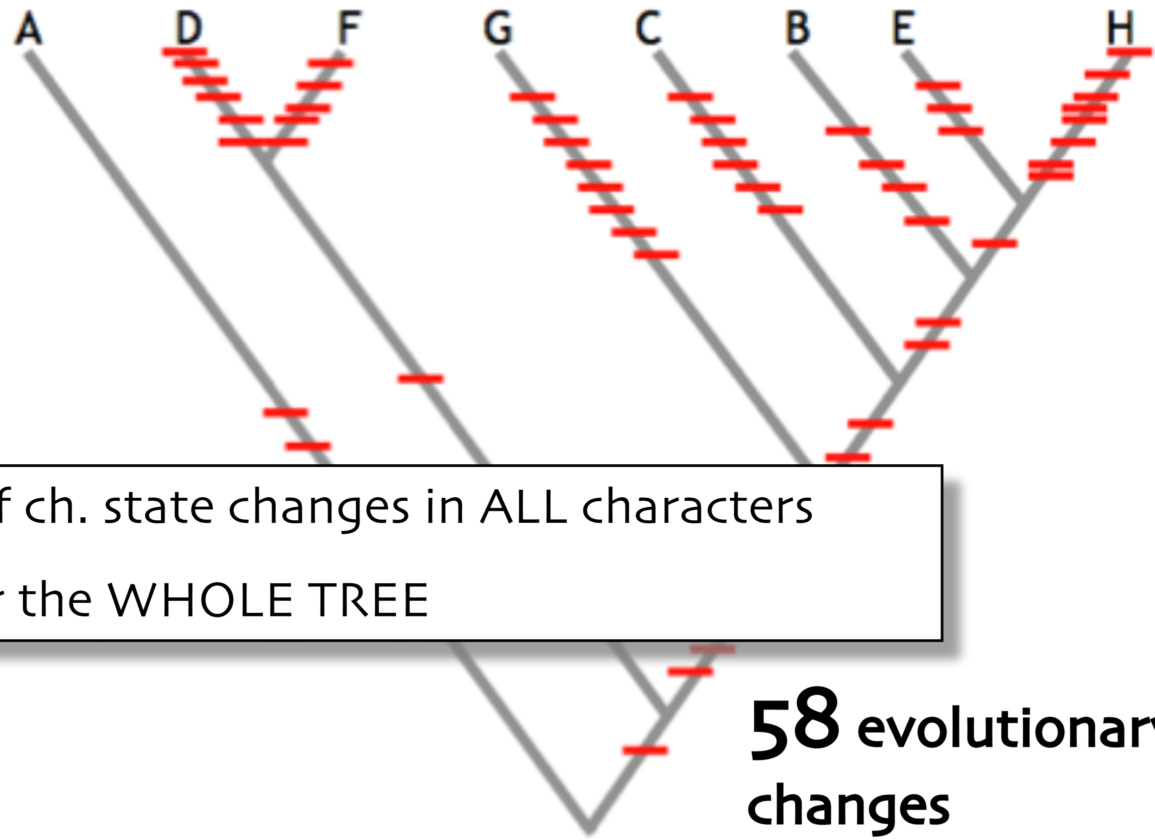
$$r = \frac{g - s}{g - m}$$

$$R = \frac{\sum g - \sum s}{\sum g - \sum m}$$

ATTENTION ! it might be HIGHLY informative to calculate indices also LOCALLY, i.e. for certain clades



38 evolutionary changes



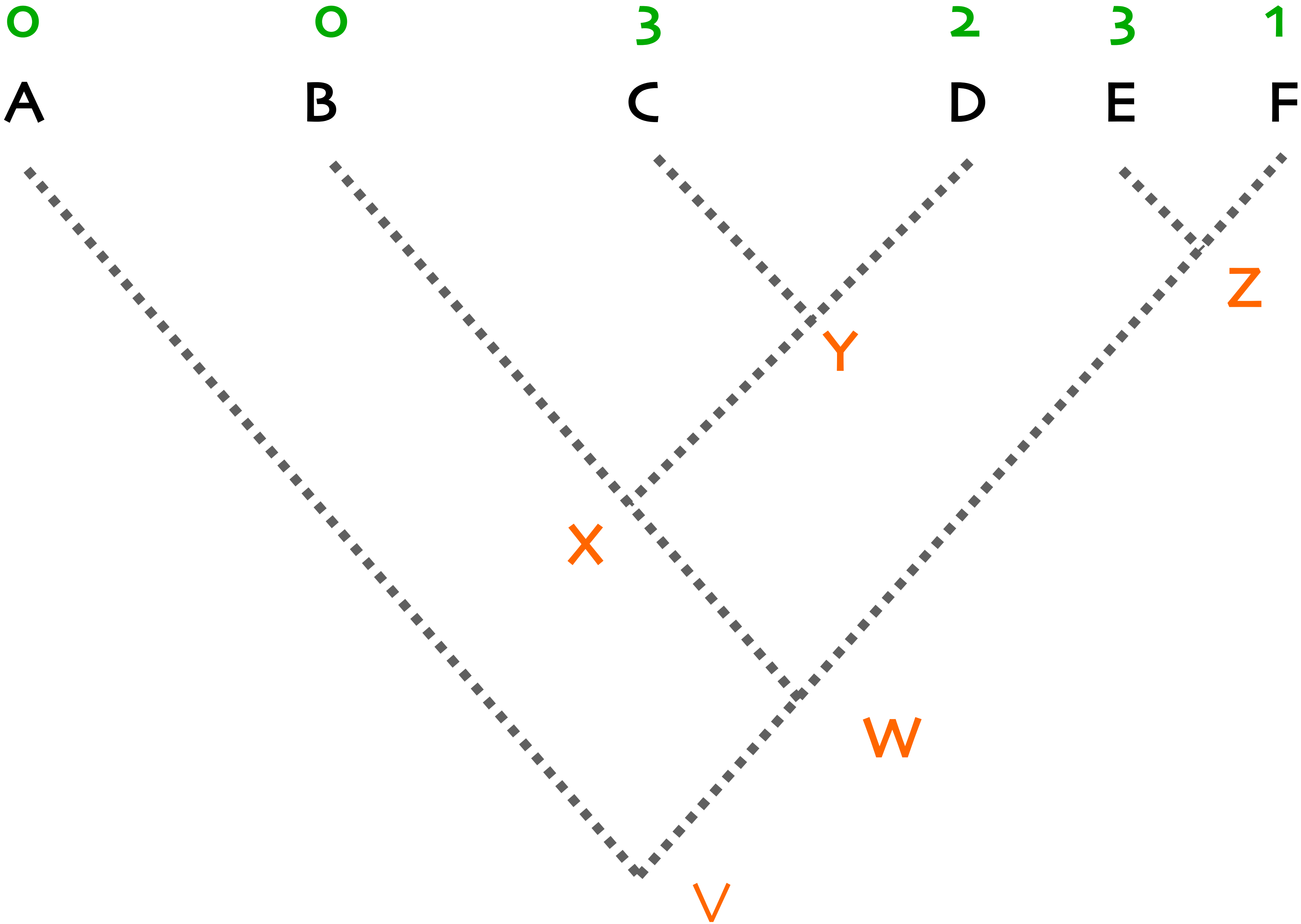
TREE LENGTH (L) = number of ch. state changes in ALL characters over the WHOLE TREE

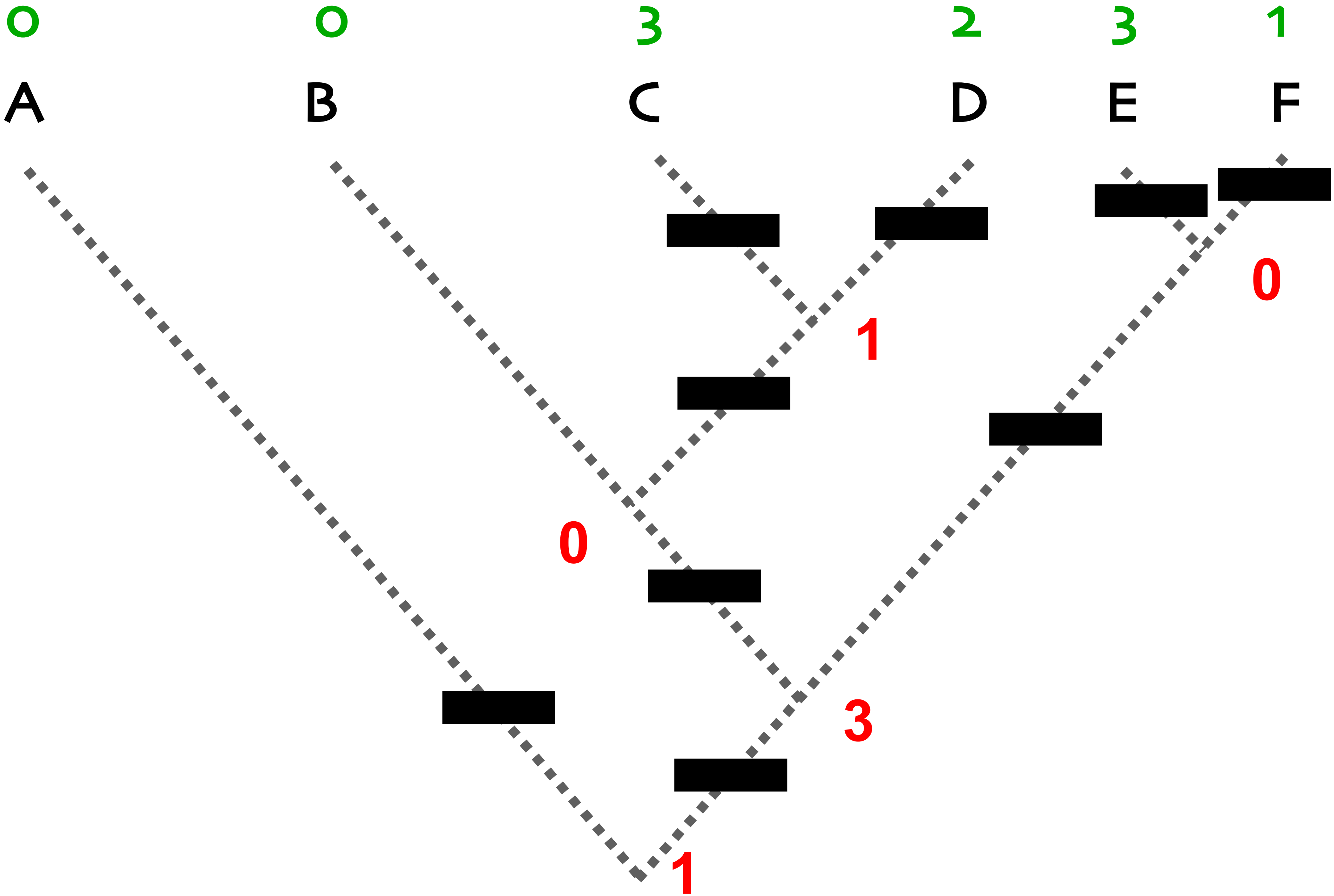
58 evolutionary changes

OPTIMIZATION

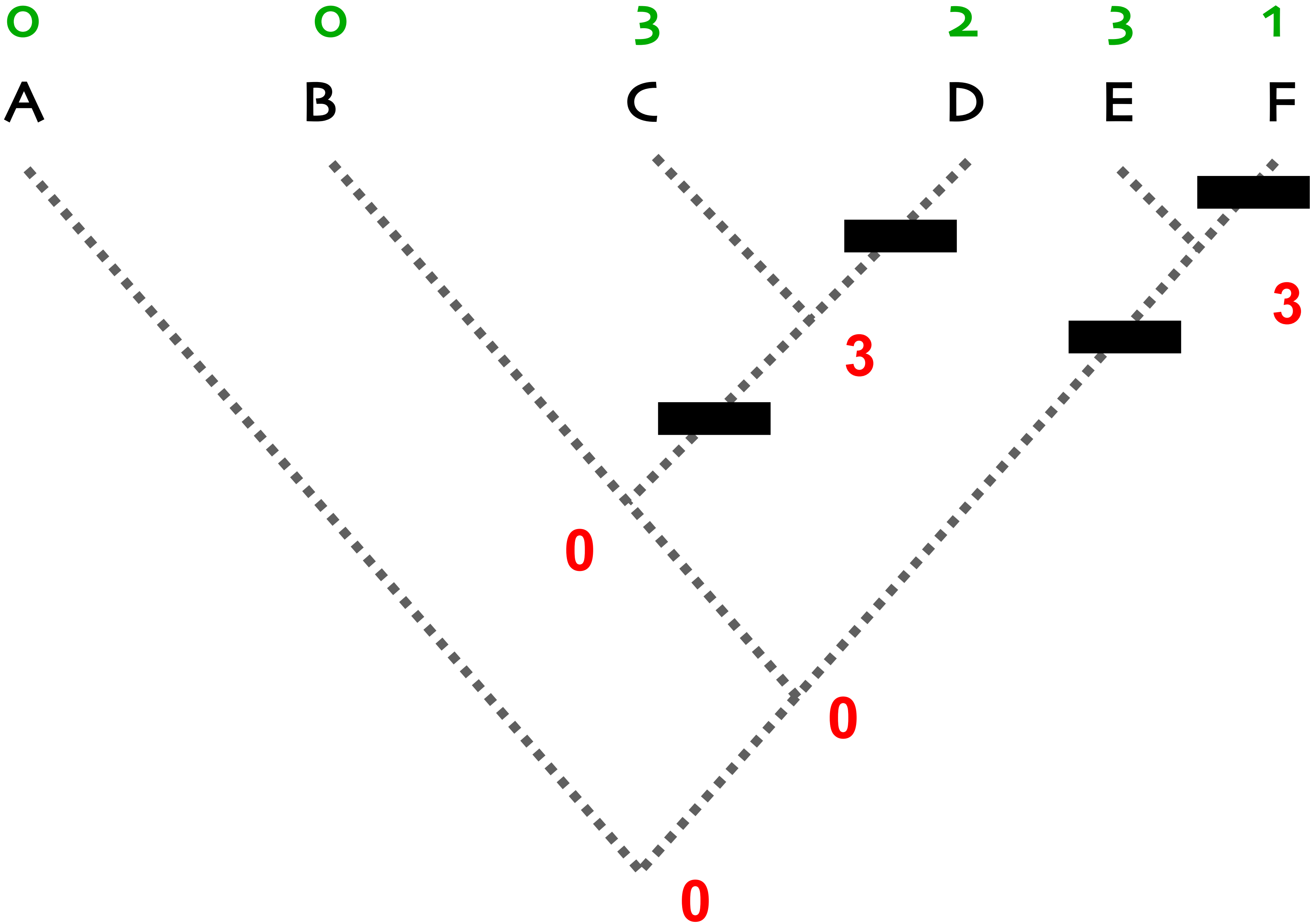
HTU, Hypothetical Taxonomic Unit

reconstruction of character states for internal nodes (HTU) of tree


0
0
3
2
3
1
A
B
C
D
E
F
Z
Y
X
W
V



0
0
3
2
3
1
A
B
C
D
E
F
0
1
0
3
1


0
0
3
2
3
1
A
B
C
D
E
F
3
3
0
0
0

OPTIMIZATION

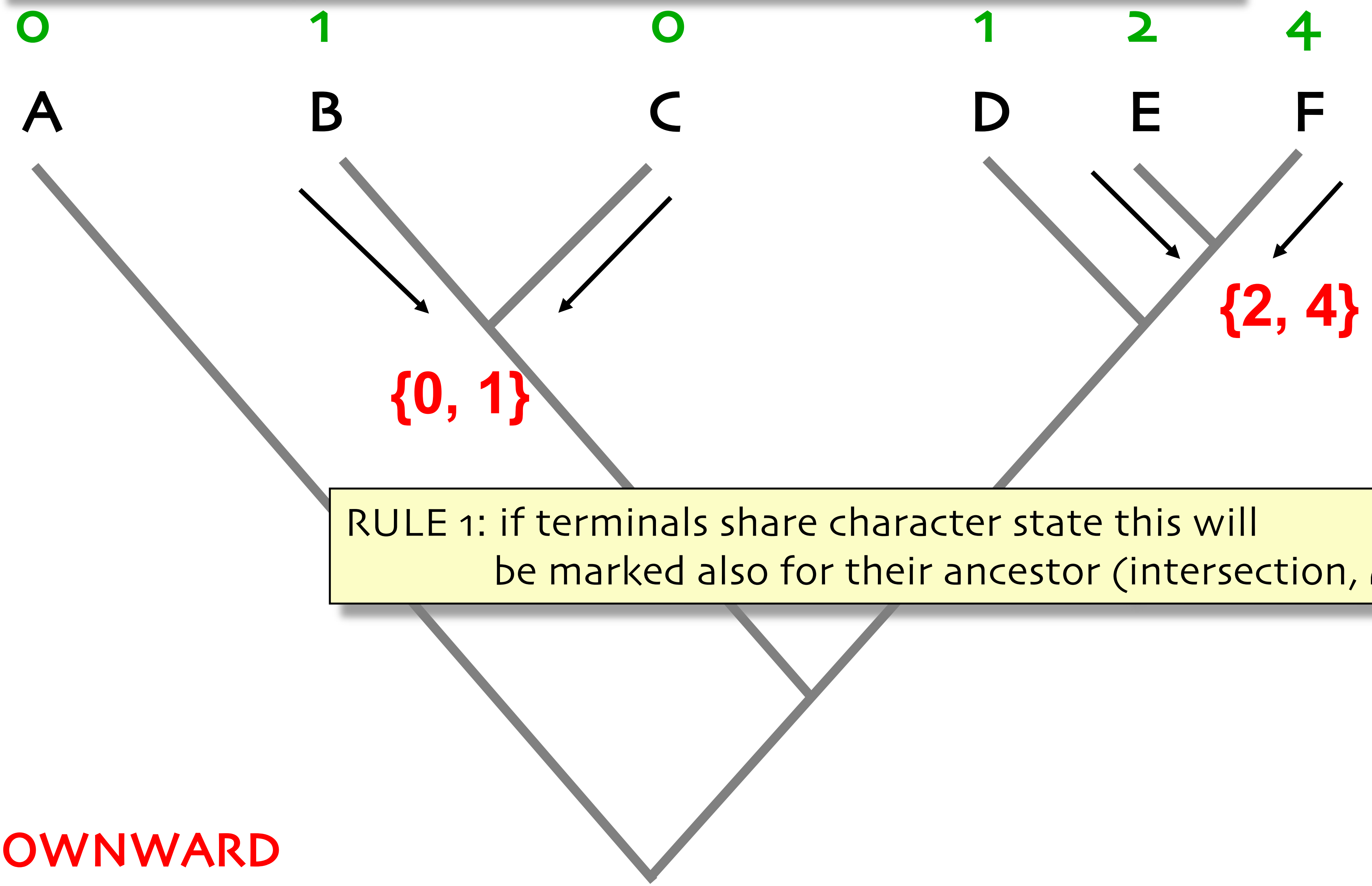
Farris, J.S. 1970. *Methods for computing Wagner trees.*

Systematic Zoology 19: 83-92.

Fitch, W.M. 1971. Toward defining the course of evolution : minimal change for a specific tree topology.

Systematic Zoology 20: 406-416.

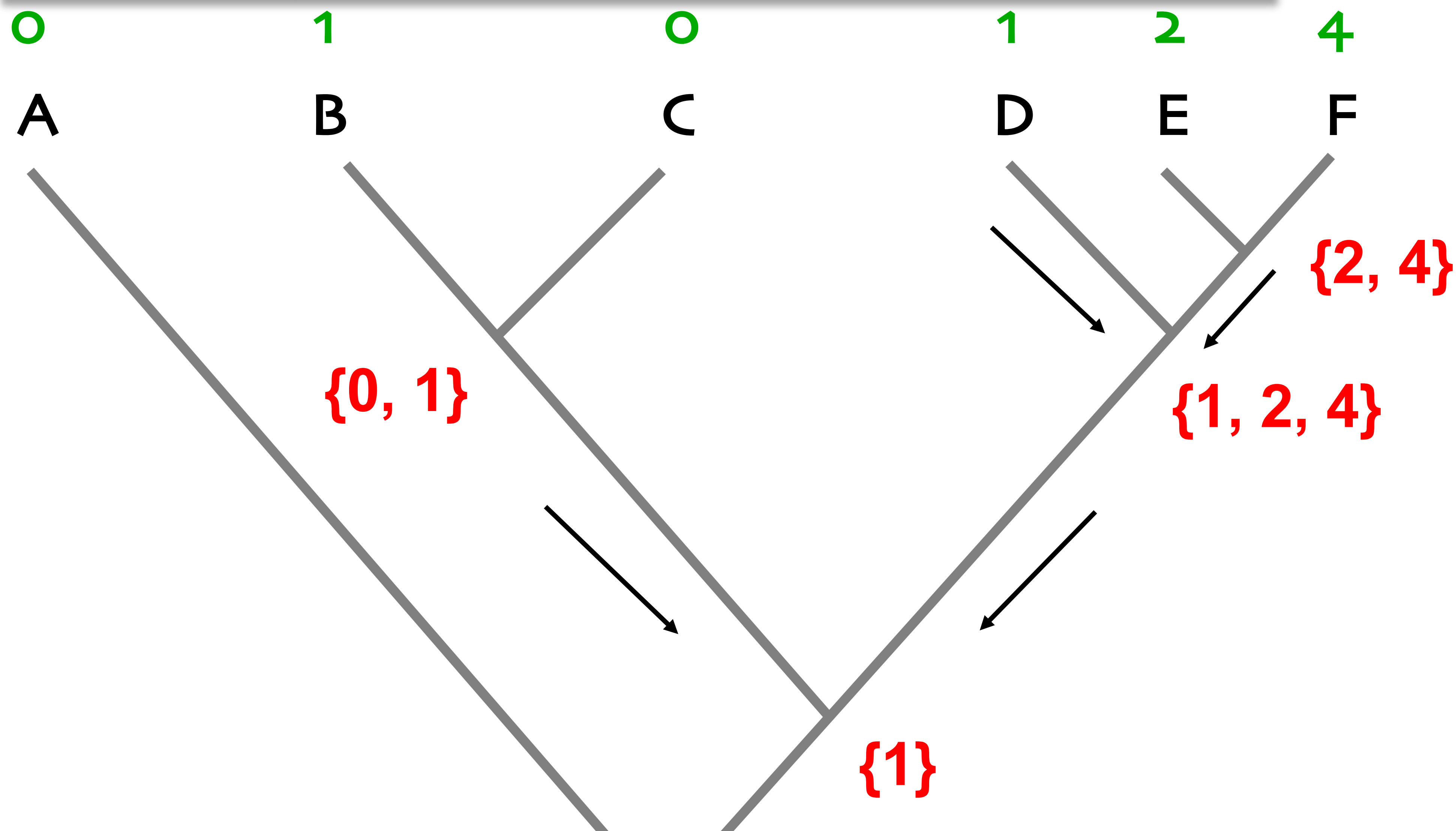
RULE 2: if terminals do not share ch. state (intersection, $\cap = \emptyset$) their (union, $\cup$) is marked for their ancestor



RULE 1: if terminals share character state this will be marked also for their ancestor (intersection, $\cap$)

DOWNWARD PASS = postorder

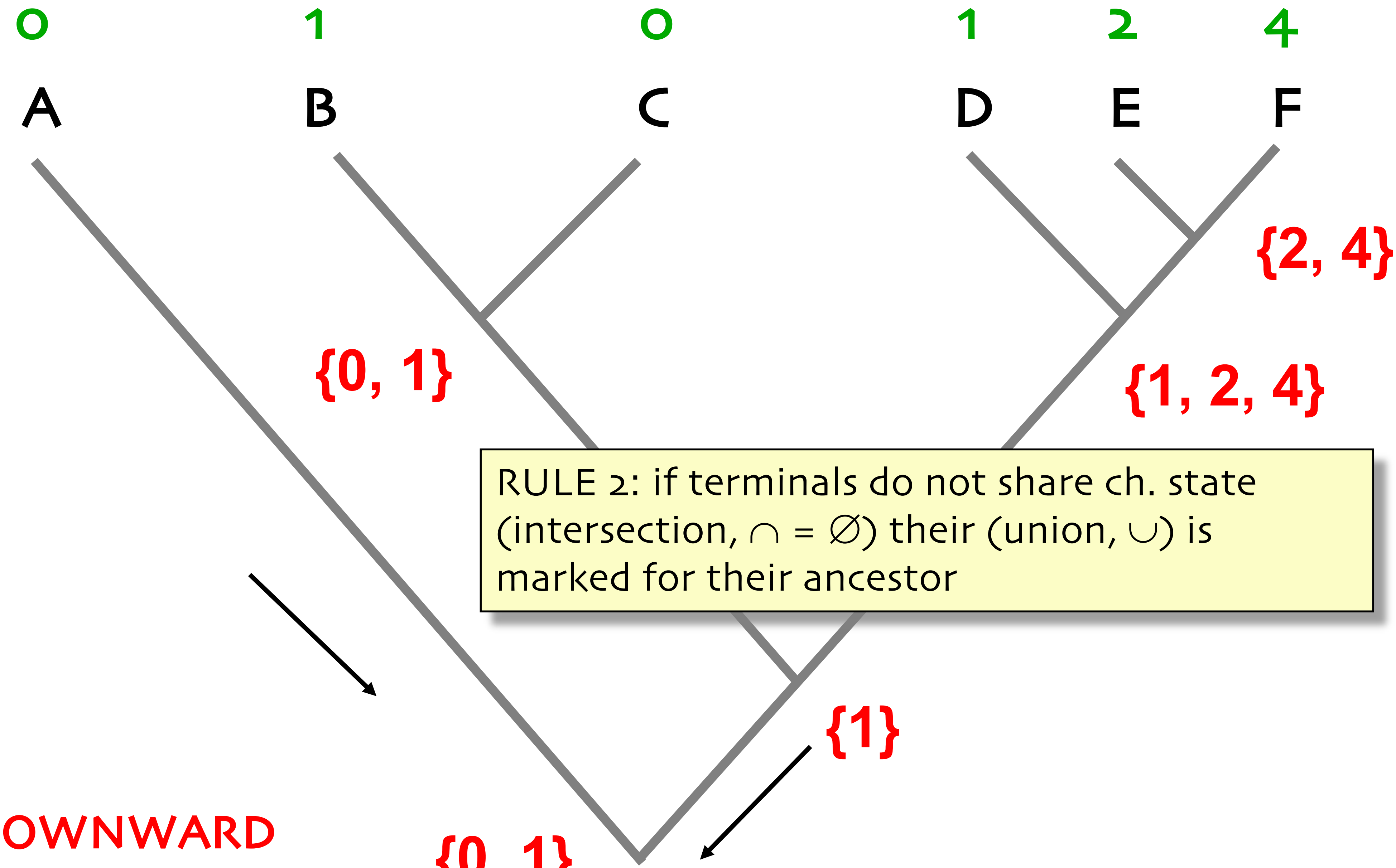
RULE 2: if terminals do not share ch. state (intersection, $\cap = \varnothing$) their (union, $\cup$) is marked for their ancestor



DOWNWARD PASS

RULE 1: if terminals share character state this will be marked also for their ancestor (intersection, $\cap$)

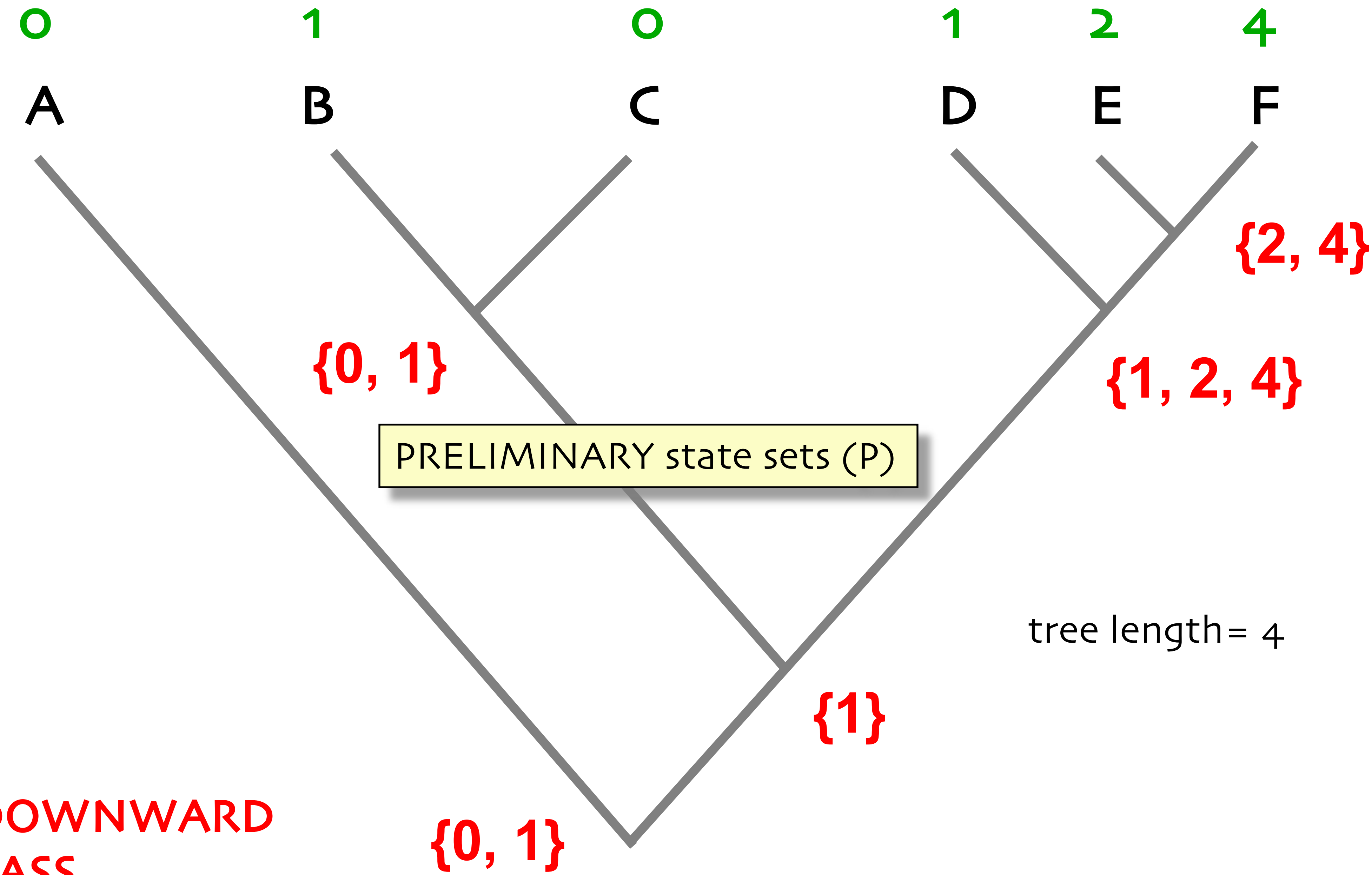
RULE 1: if terminals share character state this will be marked also for their ancestor (intersection, ∩)



RULE 2: if terminals do not share ch. state (intersection, ∩ = ∅) their (union, ∪) is marked for their ancestor

DOWNWARD PASS

ATTENTION! LENGTH of diagram, number of ch. state changes, calculated already at this stage. Unions (∪) add always one ch. state change.



DOWNWARD PASS

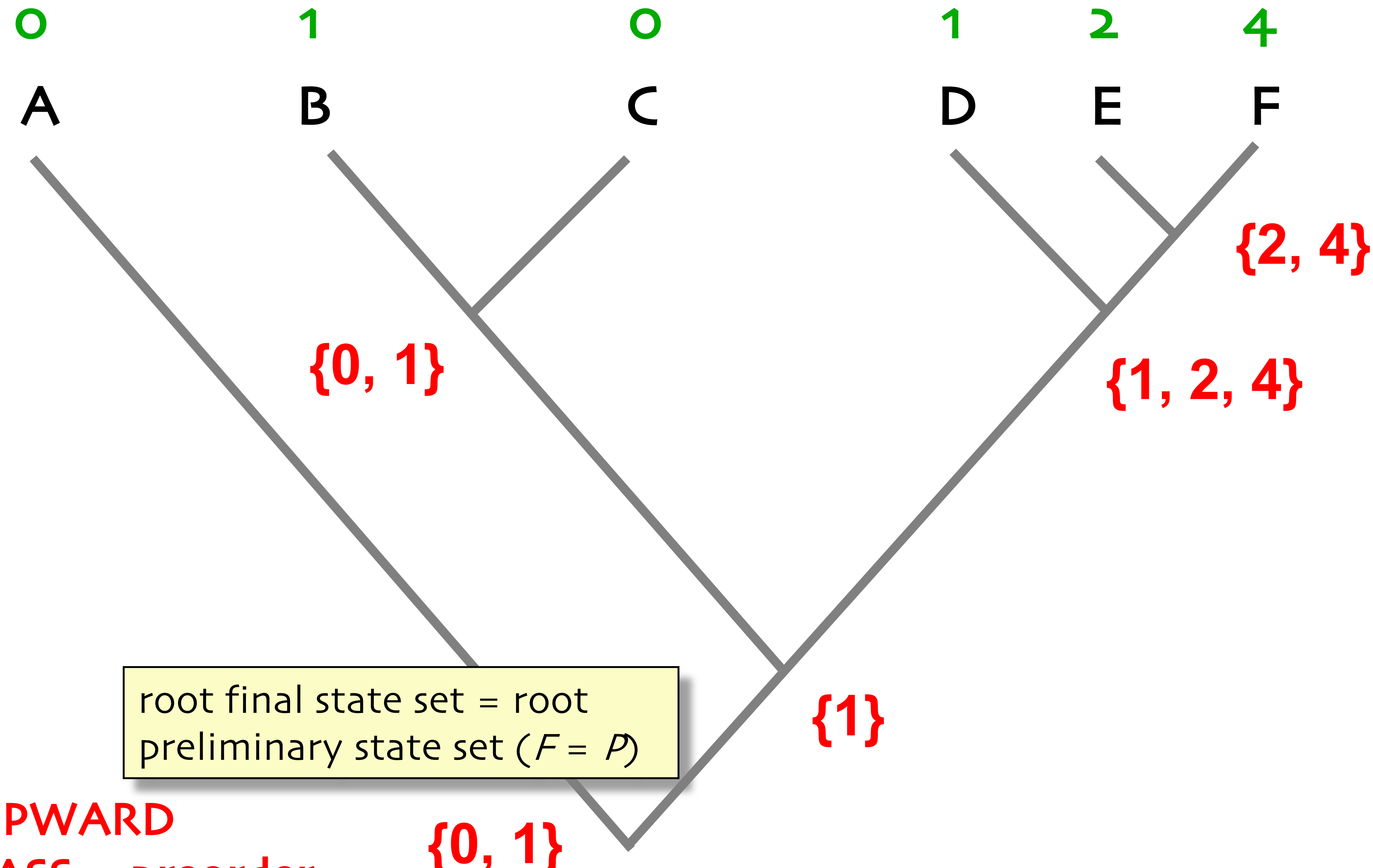
" upward pass" rules (Fitch 1971, Wheeler 2012)

- root final state set = root preliminary state set ($F = P$)

RULE 1. If the overlap of the preliminary state, P, of the node and its ancestor, A, is equal to A, (if $A \cap P = A$) then the final state set, F, is equal to that of the ancestor ($F = A$).

RULE 2. If Rule 1 does not apply and the union of final/preliminary states of the 2 descendants of the current node (*Left* and *Right*) are equal to preliminary states of the current node ($P = L \cup R$), then $F = P \cup A$.

RULE 3. If Rule 1 and 2 do not apply the final state set is the preliminary state set, supplemented by state set that is common to the ancestor and descendants ($F = P \cup (L \cap A) \cup (R \cap A)$).

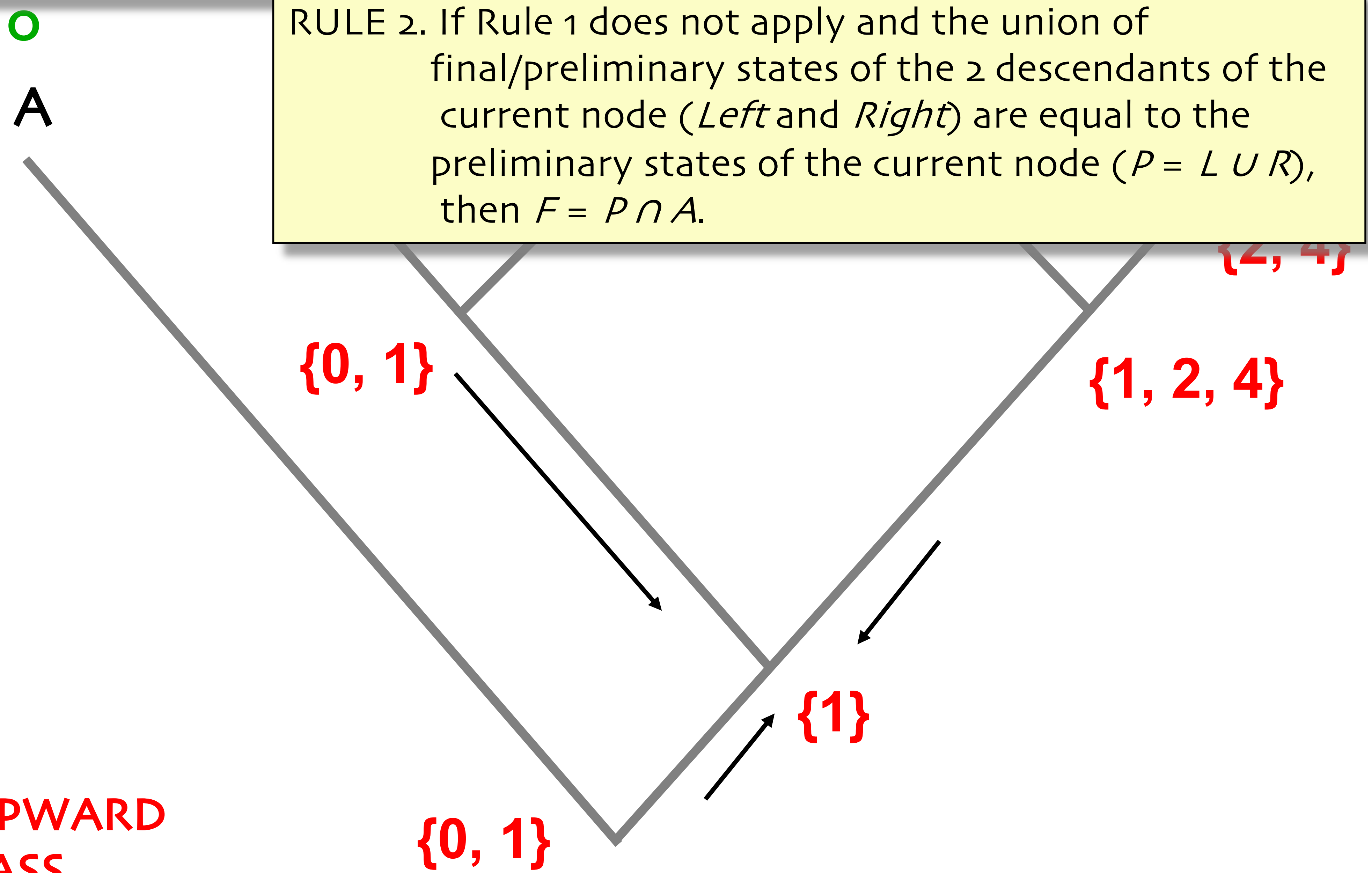


root final state set = root preliminary state set ($F = P$)

UPWARD PASS = preorder

RULE 1. If the overlap of the preliminary state, *P*, of the node and its ancestor, *A*, is equal to *A*, (if $A \cap P = A$) then the final state set, *F*, is equal to that of the ancestor ($F = A$).

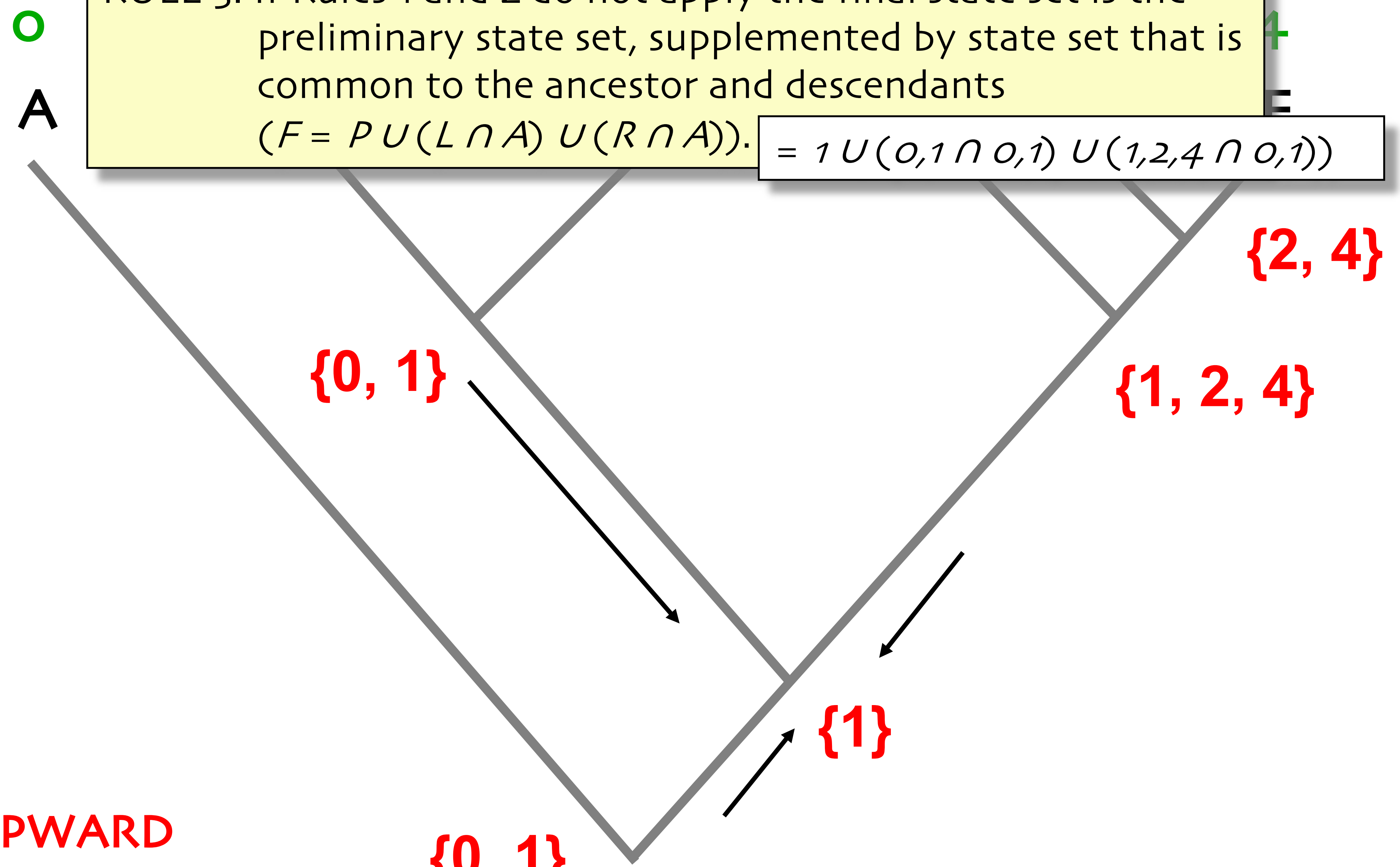
RULE 2. If Rule 1 does not apply and the union of final/preliminary states of the 2 descendants of the current node (*Left* and *Right*) are equal to the preliminary states of the current node ($P = L \cup R$), then $F = P \cap A$.



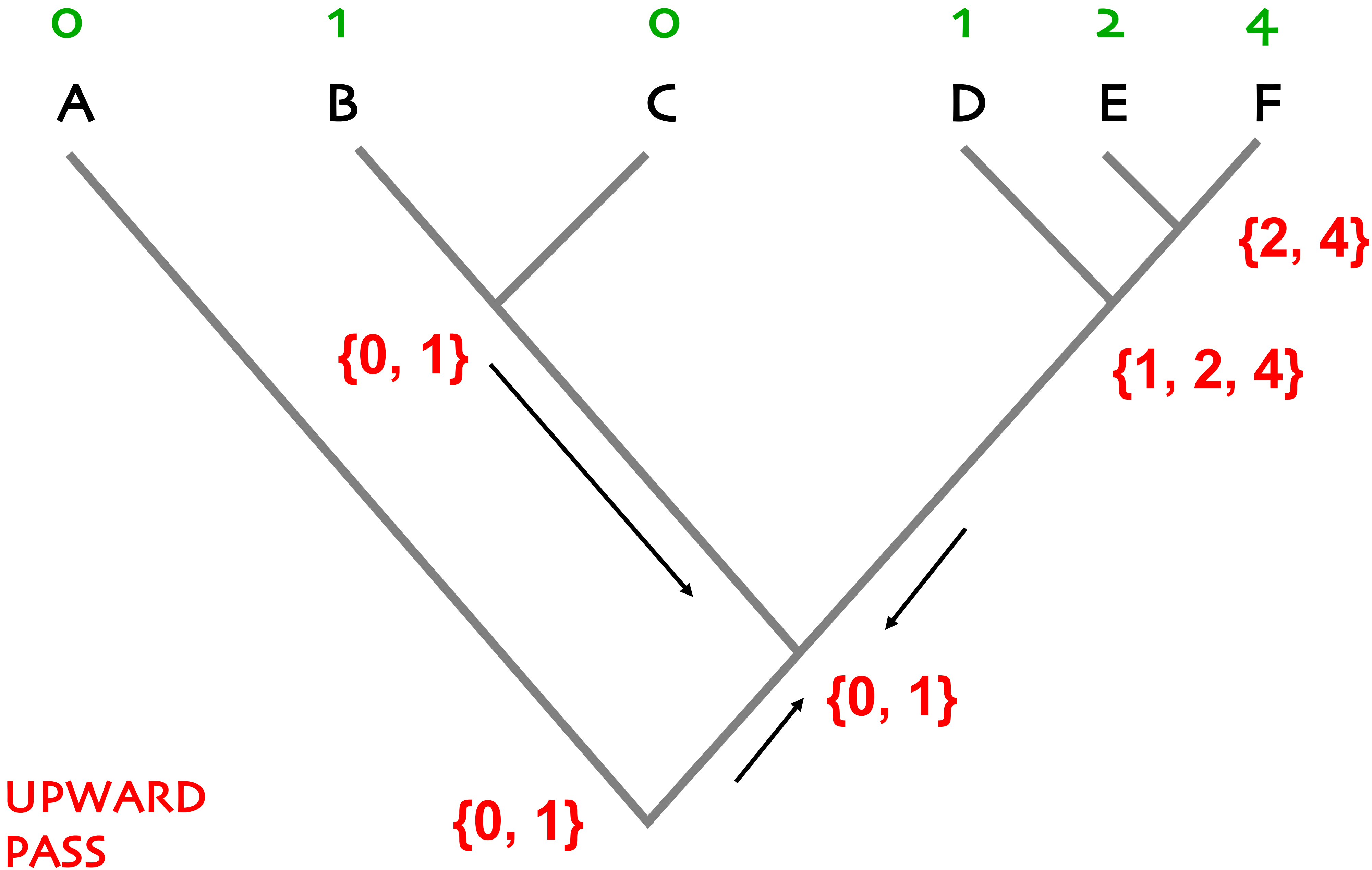
UPWARD PASS

RULE 3. If Rules 1 and 2 do not apply the final state set is the preliminary state set, supplemented by state set that is common to the ancestor and descendants ($F = P \cup (L \cap A) \cup (R \cap A)$).

$= 1 \cup (0,1 \cap 0,1) \cup (1,2,4 \cap 0,1))$

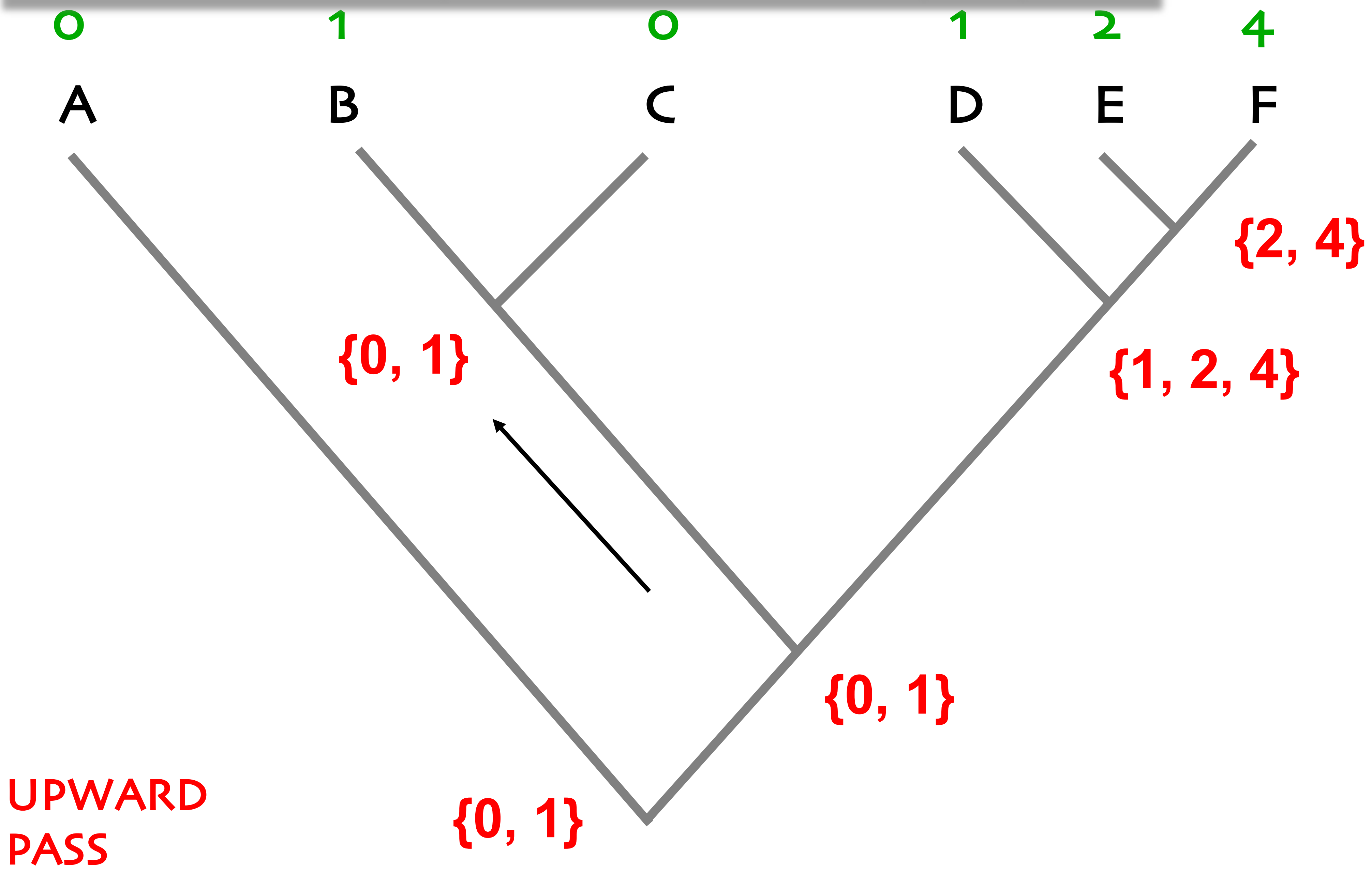


UPWARD PASS

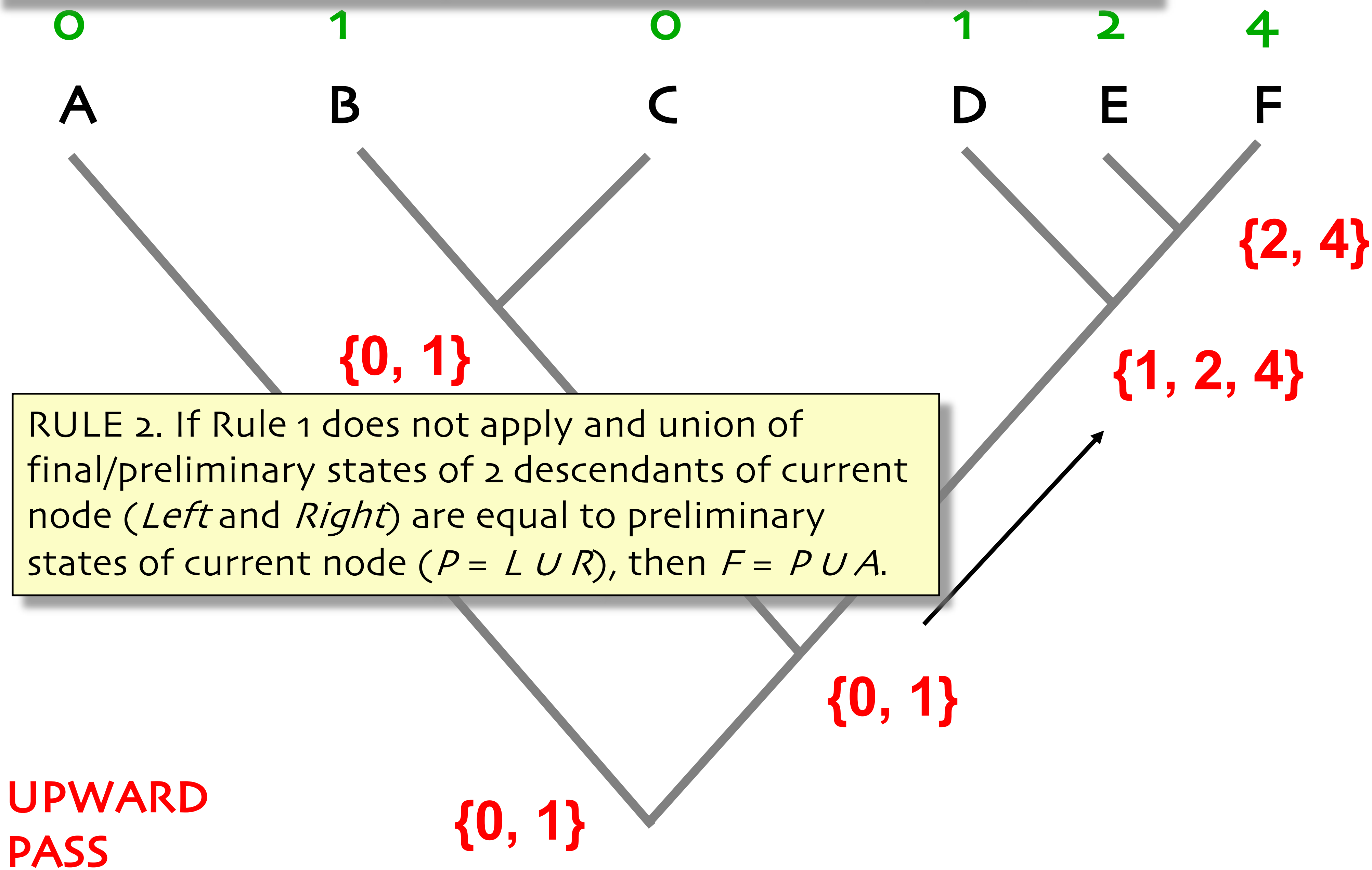


0
1
0
1
2
4
A
B
C
D
E
F
{2, 4}
{0, 1}
{1, 2, 4}
{0, 1}
{0, 1}
UPWARD
PASS

RULE 1. If the overlap of the preliminary state, *P*, of the node and its ancestor, *A*, is equal to *A*, (if $A \cap P = A$) then the final state set, *F*, is equal to that of the ancestor ($F = A$).



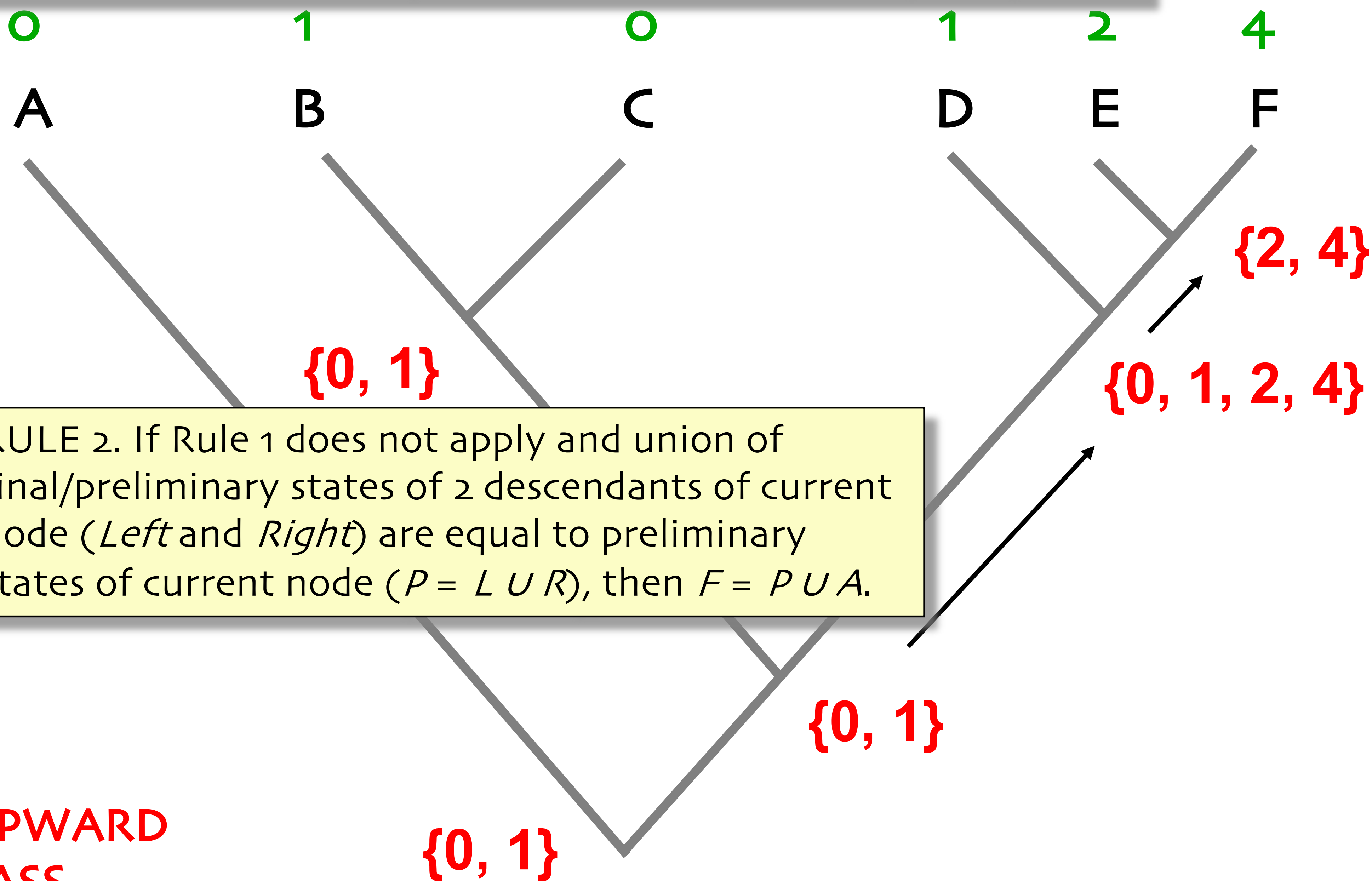
RULE 1. If the overlap of the preliminary state, *P*, of the node and its ancestor, *A*, is equal to *A*, (if $A \cap P = A$) then the final state set, *F*, is equal to that of the ancestor ($F = A$).



RULE 2. If Rule 1 does not apply and union of final/preliminary states of 2 descendants of current node (*Left* and *Right*) are equal to preliminary states of current node ($P = L \cup R$), then $F = P \cup A$.

UPWARD PASS

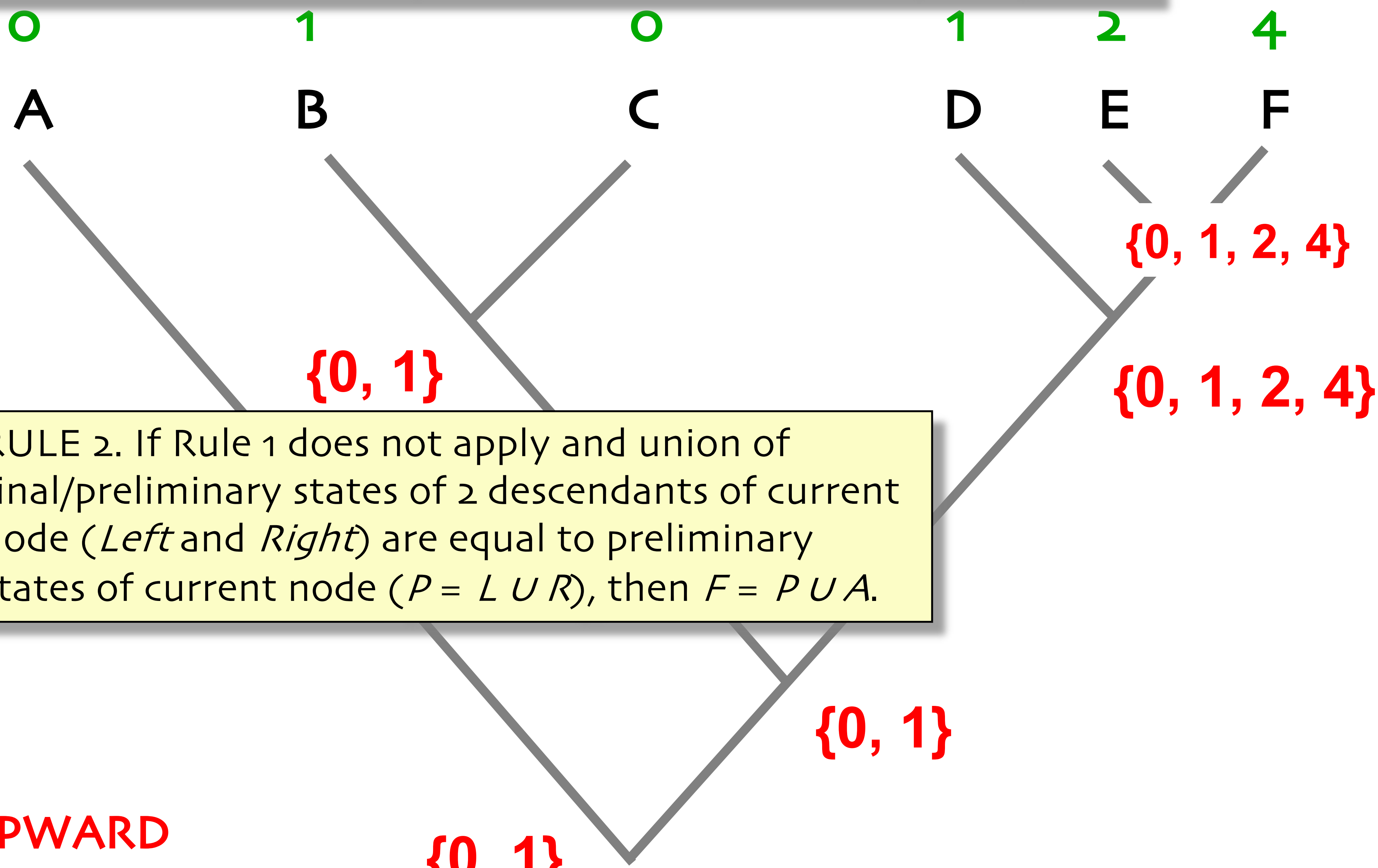
RULE 1. If the overlap of the preliminary state, *P*, of the node and its ancestor, *A*, is equal to *A*, (if $A \cap P = A$) then the final state set, *F*, is equal to that of the ancestor ($F = A$).



RULE 2. If Rule 1 does not apply and union of final/preliminary states of 2 descendants of current node (*Left* and *Right*) are equal to preliminary states of current node ($P = L \cup R$), then $F = P \cup A$.

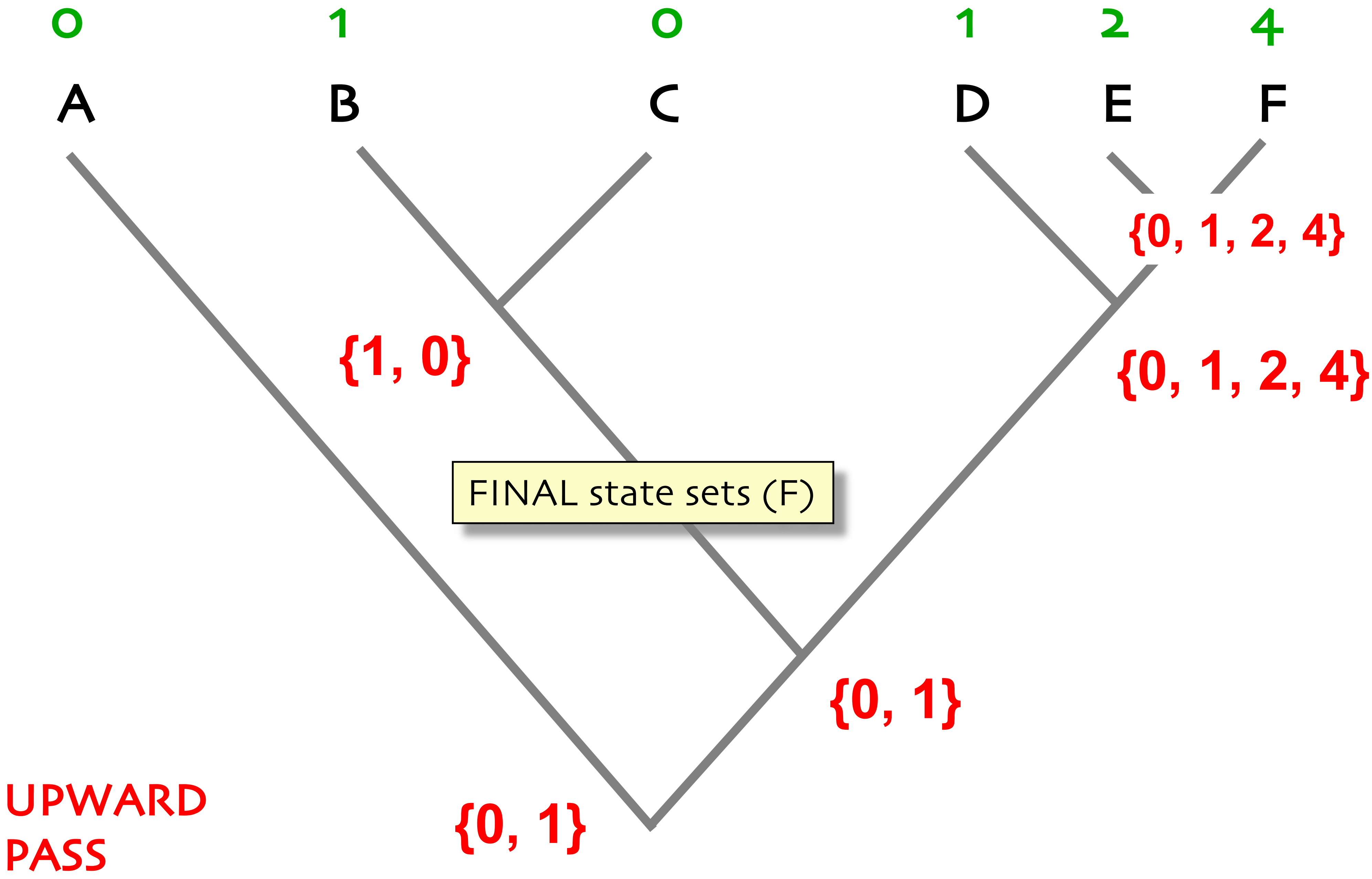
UPWARD PASS

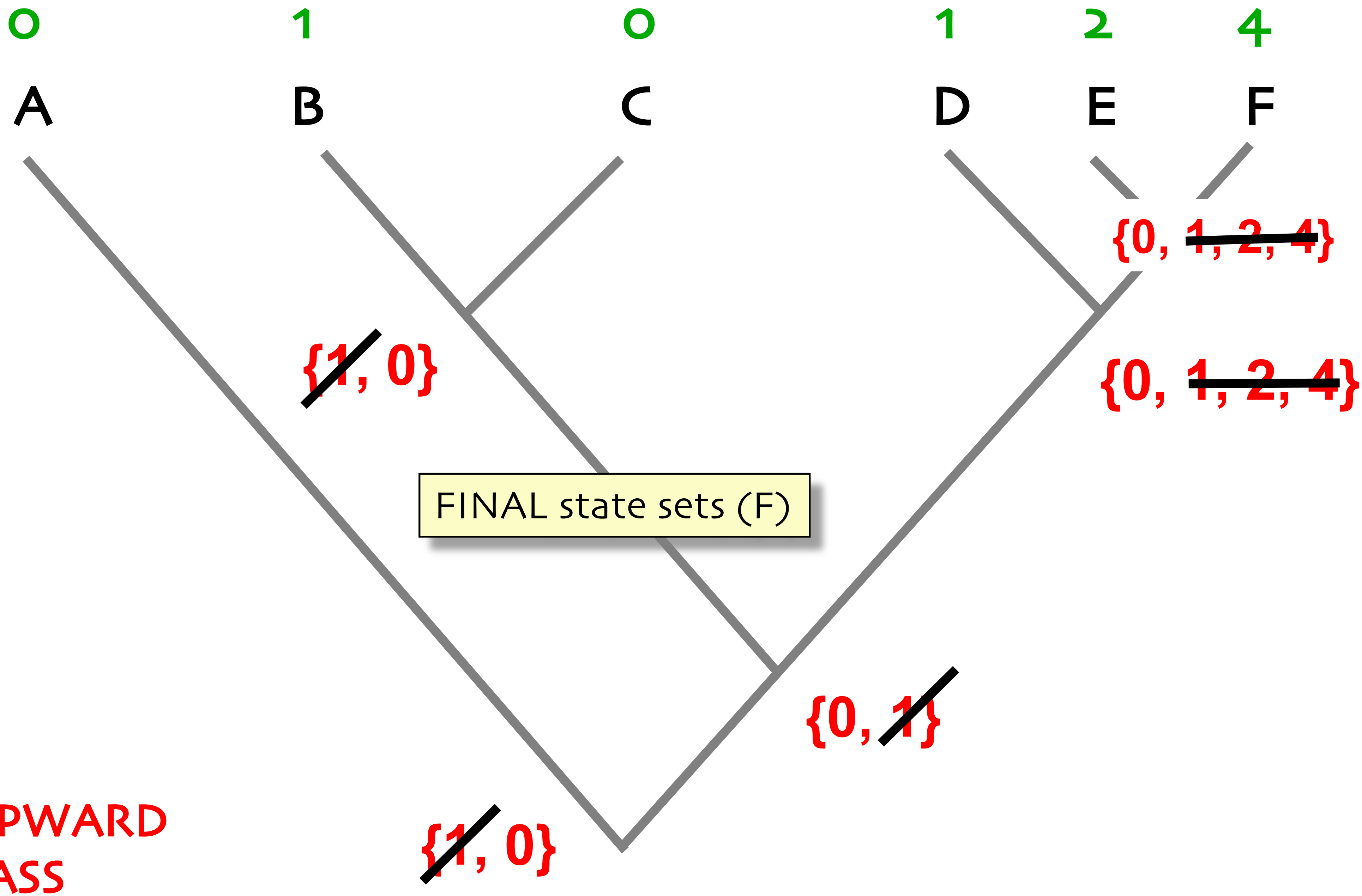
RULE 1. If the overlap of the preliminary state, *P*, of the node and its ancestor, *A*, is equal to *A*, (if $A \cap P = A$) then the final state set, *F*, is equal to that of the ancestor ($F = A$).



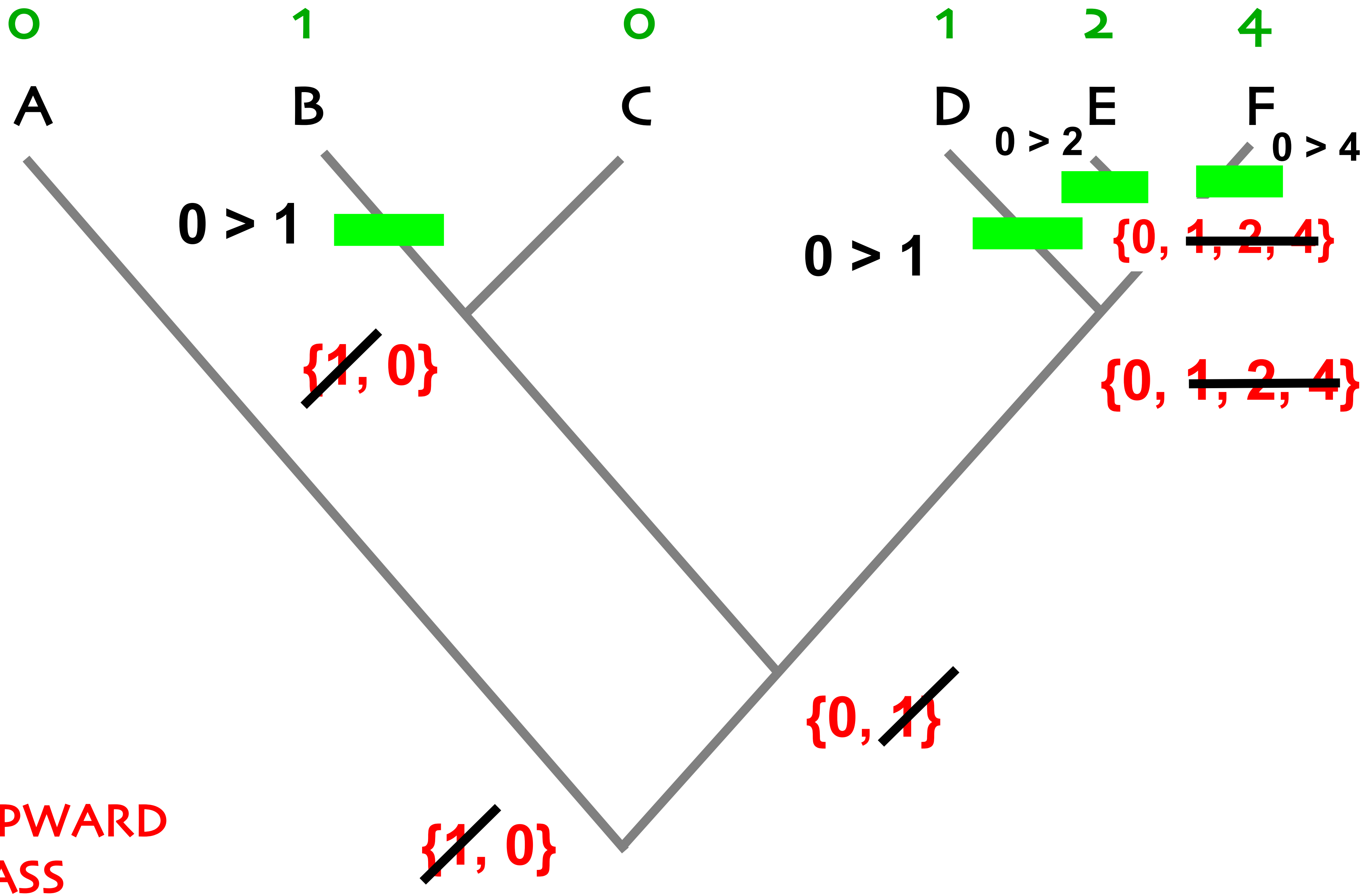
RULE 2. If Rule 1 does not apply and union of final/preliminary states of 2 descendants of current node (*Left* and *Right*) are equal to preliminary states of current node ($P = L \cup R$), then $F = P \cup A$.

UPWARD PASS

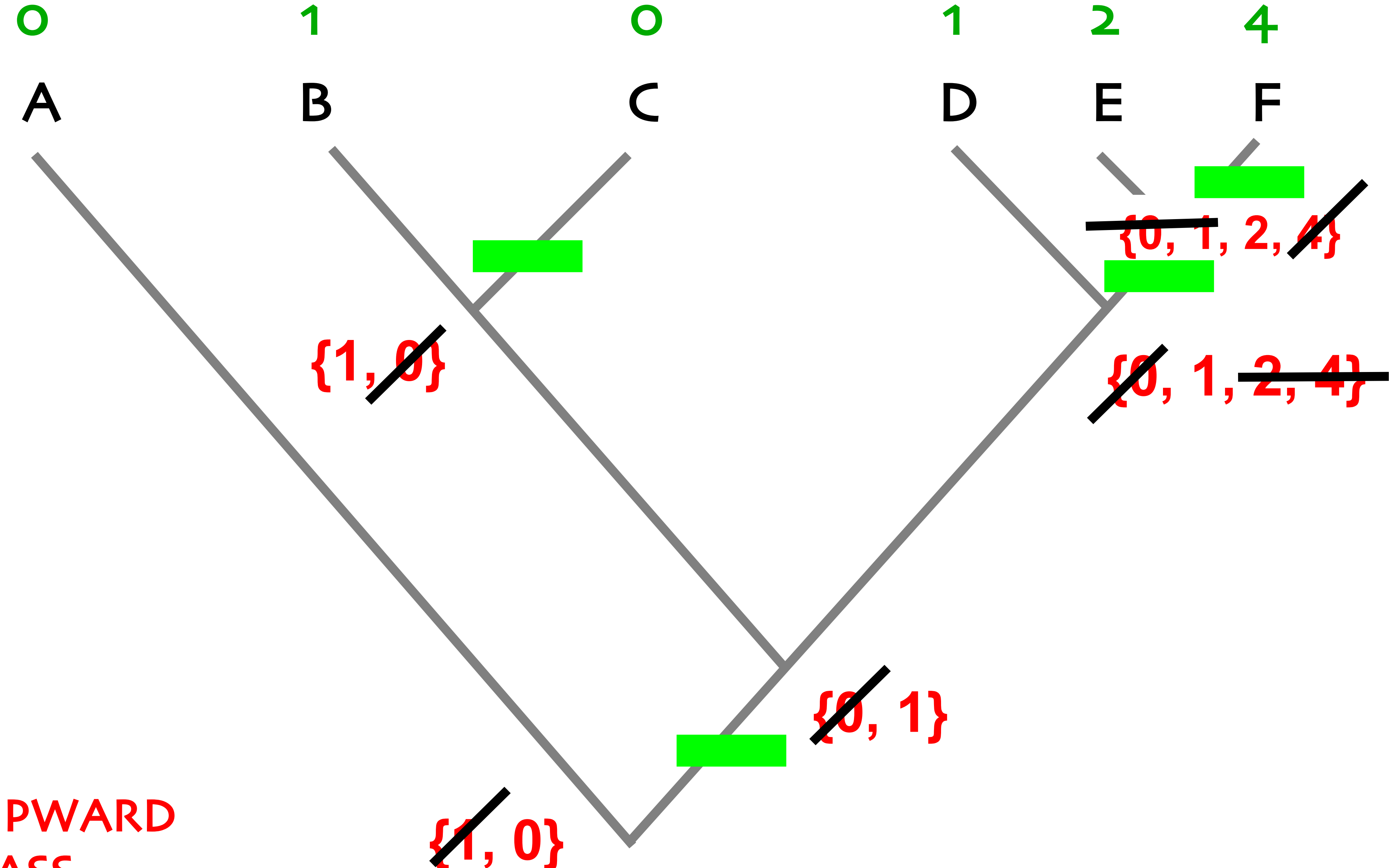

0
1
0
1
2
4
A
B
C
D
E
F
{0, 1, 2, 4}
{1, 0}
{0, 1, 2, 4}
FINAL state sets (F)
{0, 1}
UPWARD PASS
{0, 1}



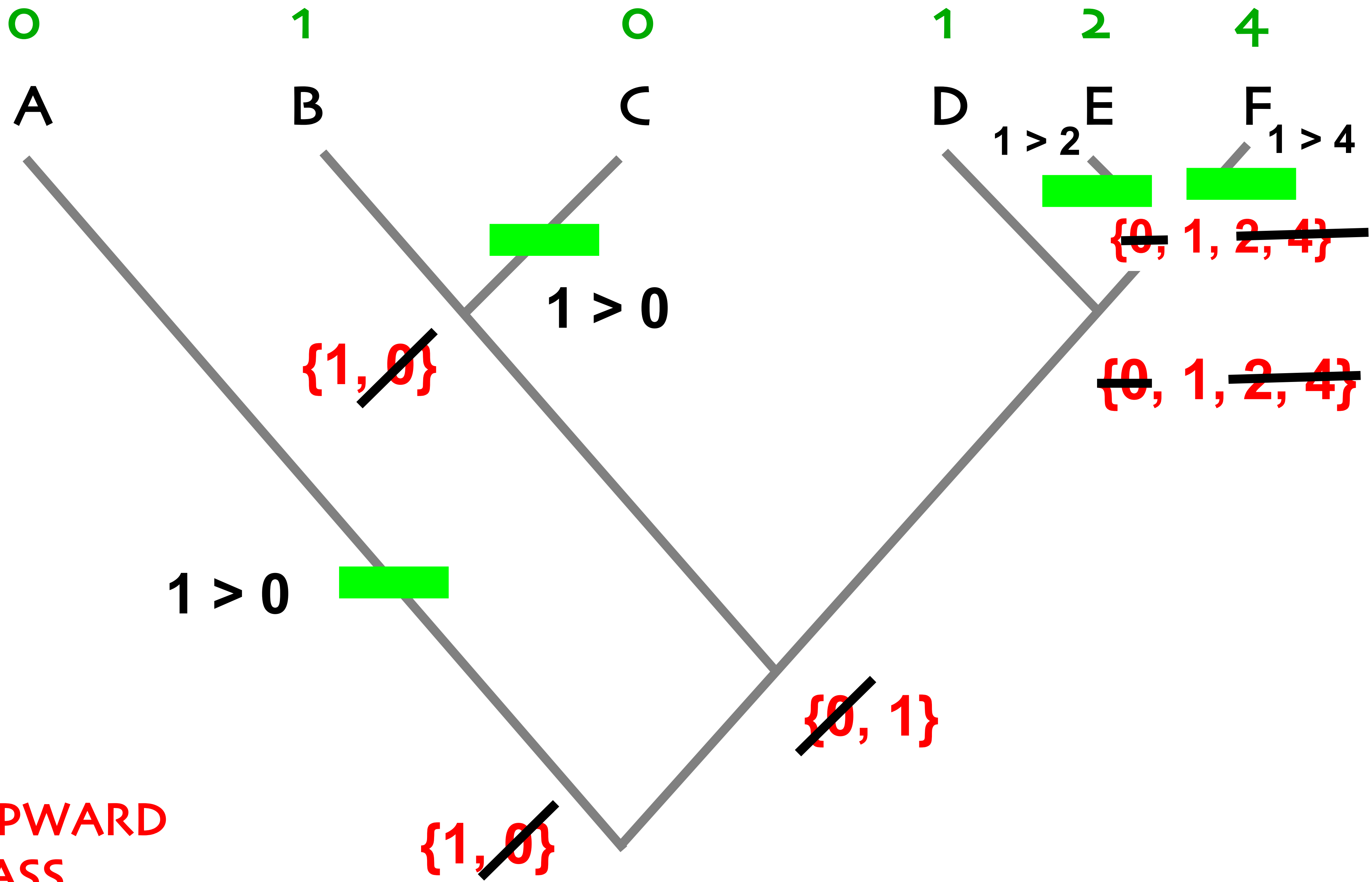
UPWARD
PASS



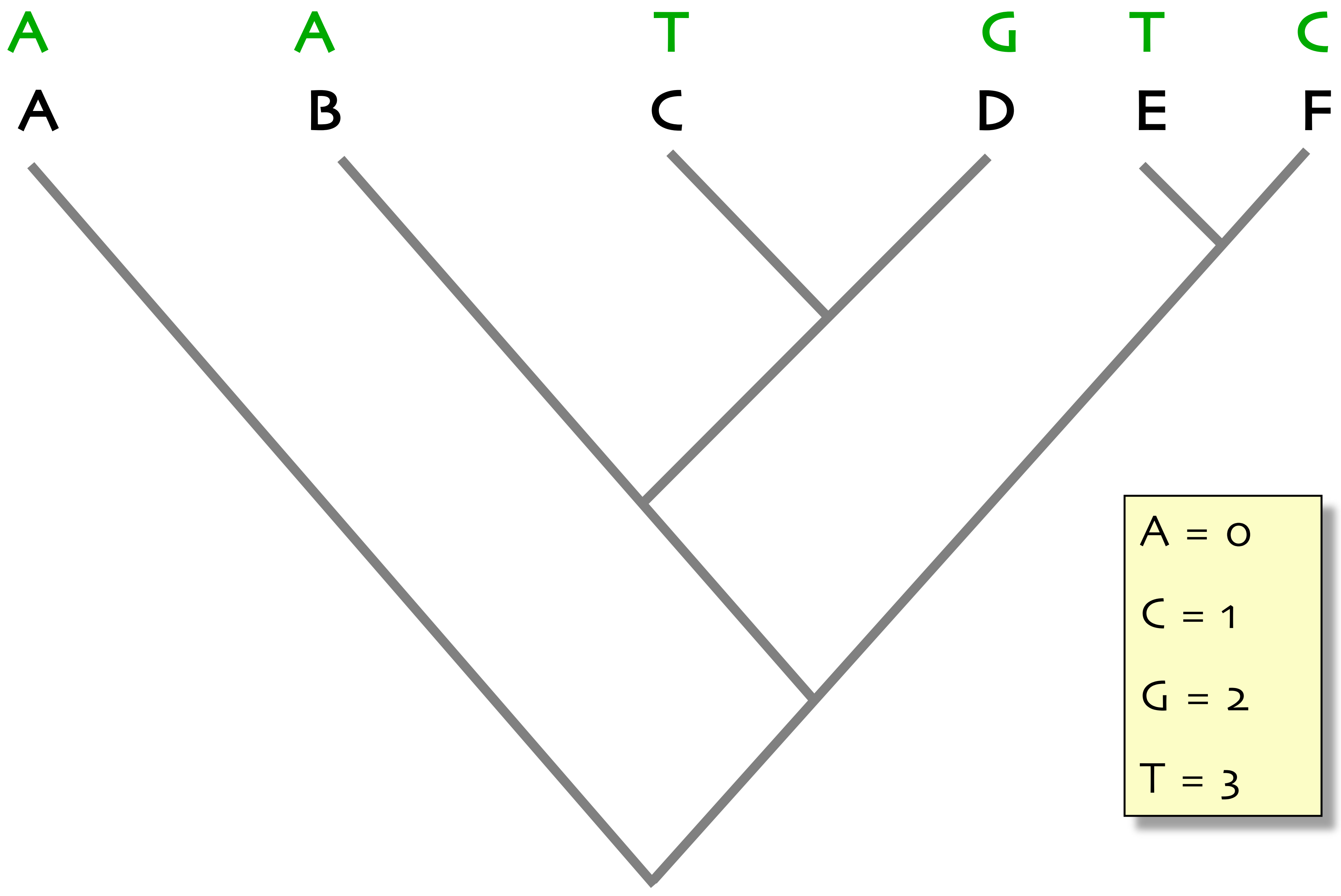
UPWARD
PASS

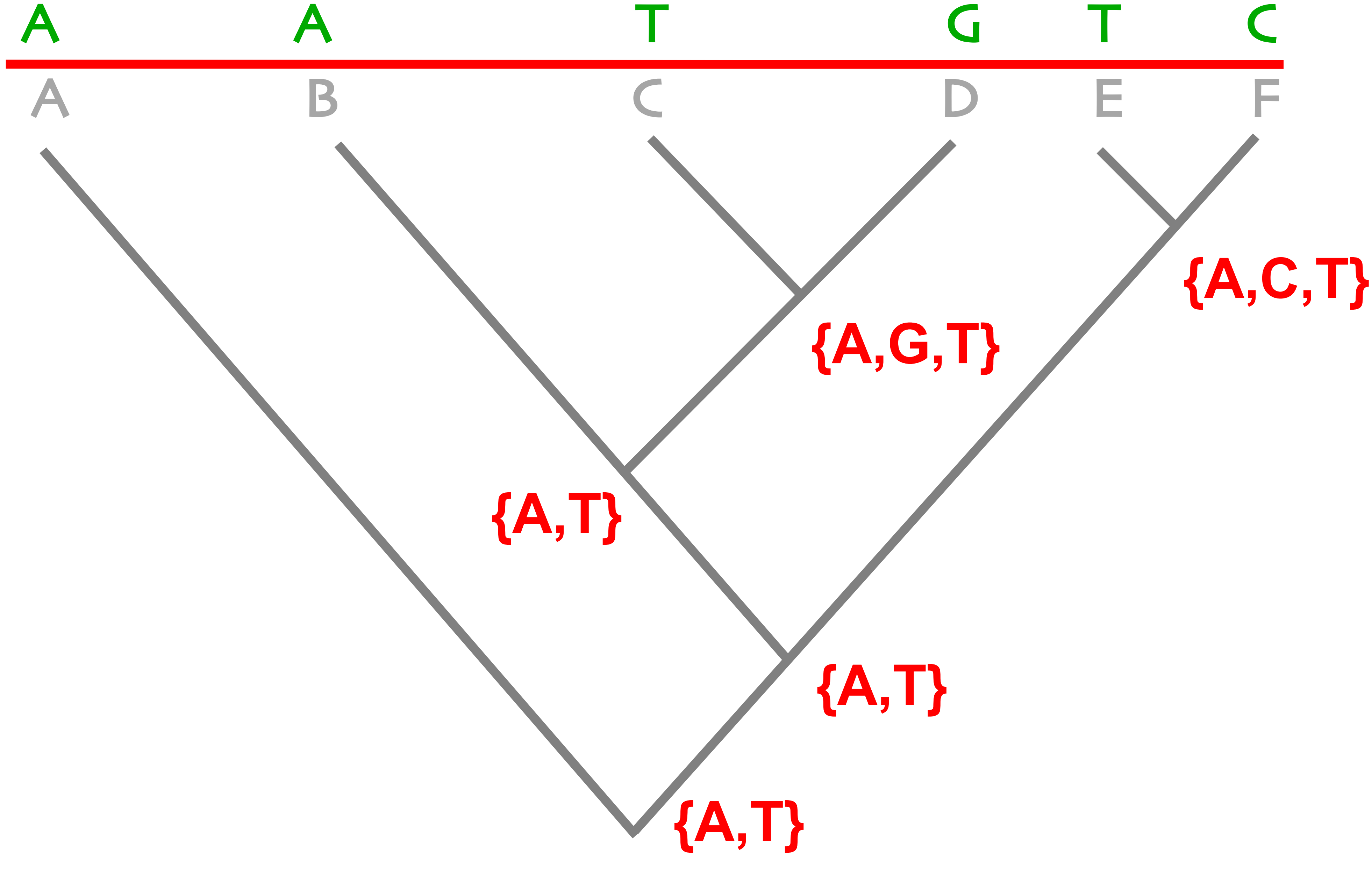


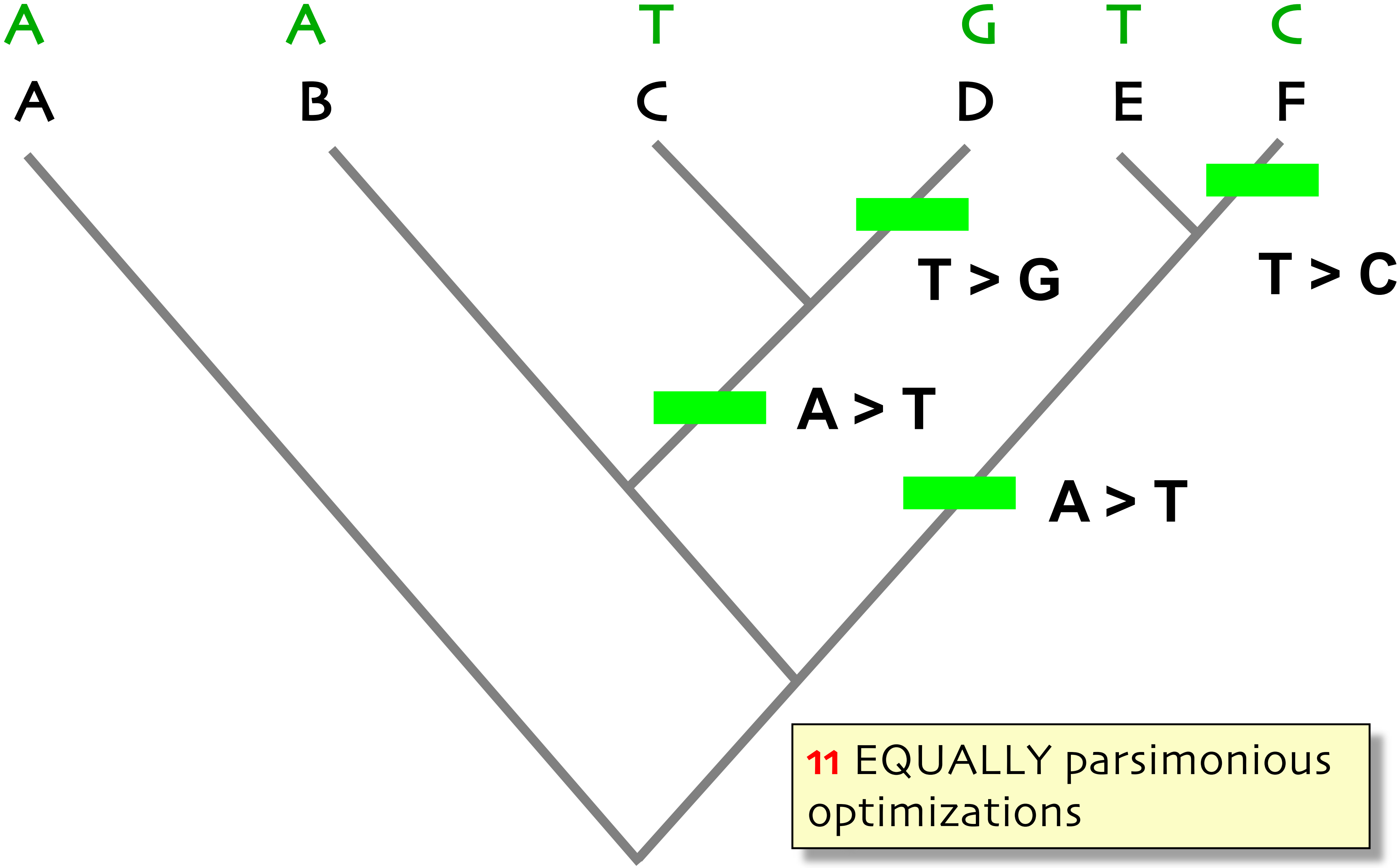
UPWARD
PASS



UPWARD PASS


A A T G T C
A B C D E F
A = 0
C = 1
G = 2
T = 3


A
A
T
G
T
C
A
B
C
D
E
F
{A,C,T}
{A,G,T}
{A,T}
{A,T}
{A,T}



11 EQUALLY parsimonious optimizations

Wagner optimization

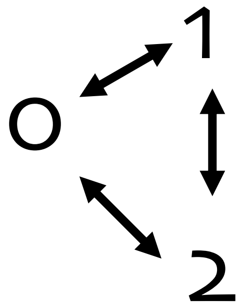
Farris, J.S. 1970. Methods for computing Wagner trees.

Systematic Zoology 19: 83-92.

Fitch, W.M. 1971. Toward defining the course of evolution : minimal change for a specific tree topology.

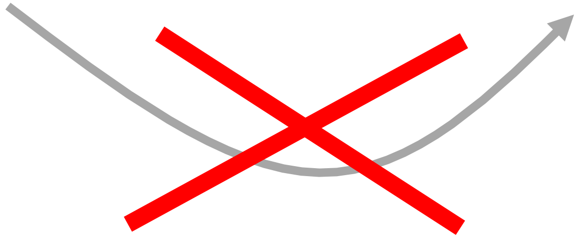
Systematic Zoology 20: 406-416.

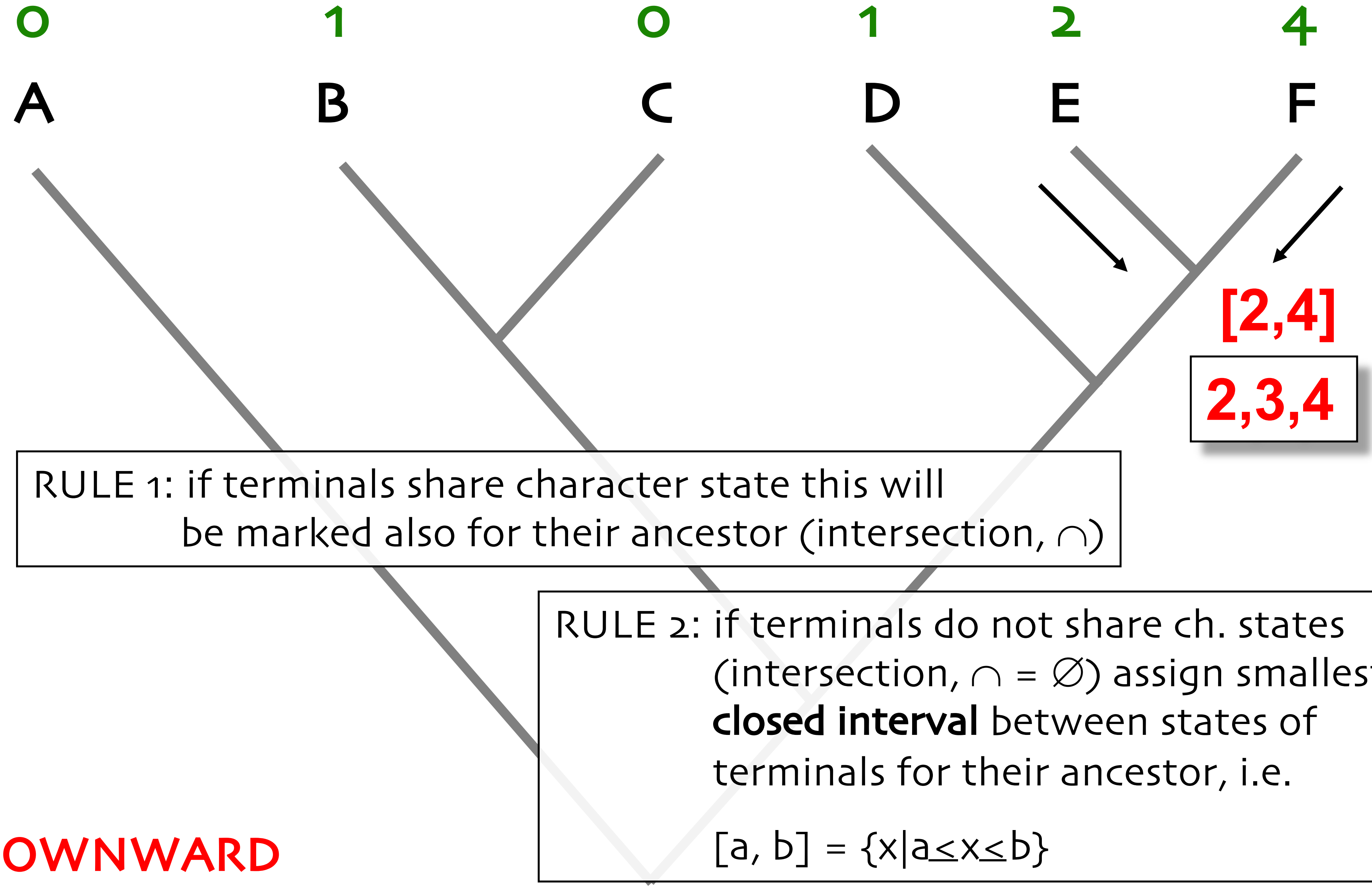
FITCH PARSIMONY



WAGNER PARSIMONY

0 <--> 1 <--> 2





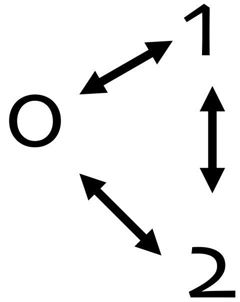
RULE 1: if terminals share character state this will be marked also for their ancestor (intersection, ∩)

RULE 2: if terminals do not share ch. states (intersection, ∩ = ∅) assign smallest **closed interval** between states of terminals for their ancestor, i.e.

$[a, b] = \{x | a \leq x \leq b\}$

DOWNWARD PASS

FITCH PARSIMONY



$A = \{0,1,2,3\}$

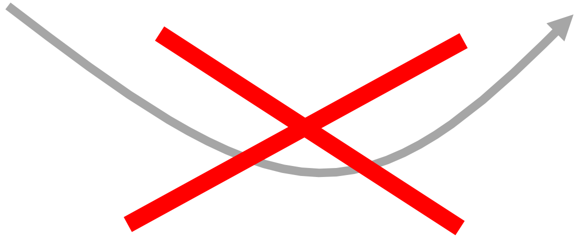
$B = \{5,6\}$

$A \cap B = \emptyset$

$A \cup B = \{0,1,2,3,4,5,6\}$

WAGNER PARSIMONY

0 <--> 1 <--> 2



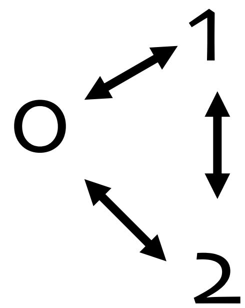
$A = \{0,1,2,3\}$

$B = \{5,6\}$

$A \cap B = \emptyset$

$A \cup B = \{3,4,5\}$

FITCH PARSIMONY



$A = \{0,1,2,3\}$

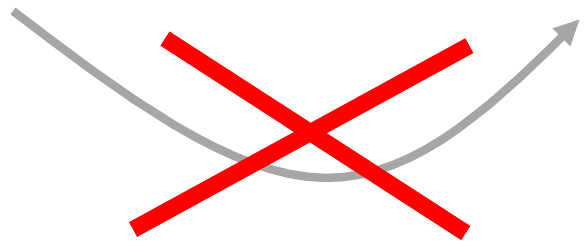
$B = \{5,6\}$

$A \cap B = \varnothing$

$A \cup B = \{0,1,2,3,4,5,6\}$

WAGNER PARSIMONY

0 <--> 1 <--> 2



$A = \{0,1,2,3\}$

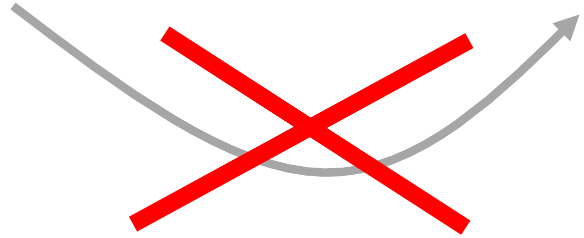
$B = \{5,6\}$

$A \cap B = \varnothing$

$A \cup B = [3,5] = \{x | 3 \leq x \leq 5\}$

WAGNER PARSIMONY

0 <--> 1 <--> 2

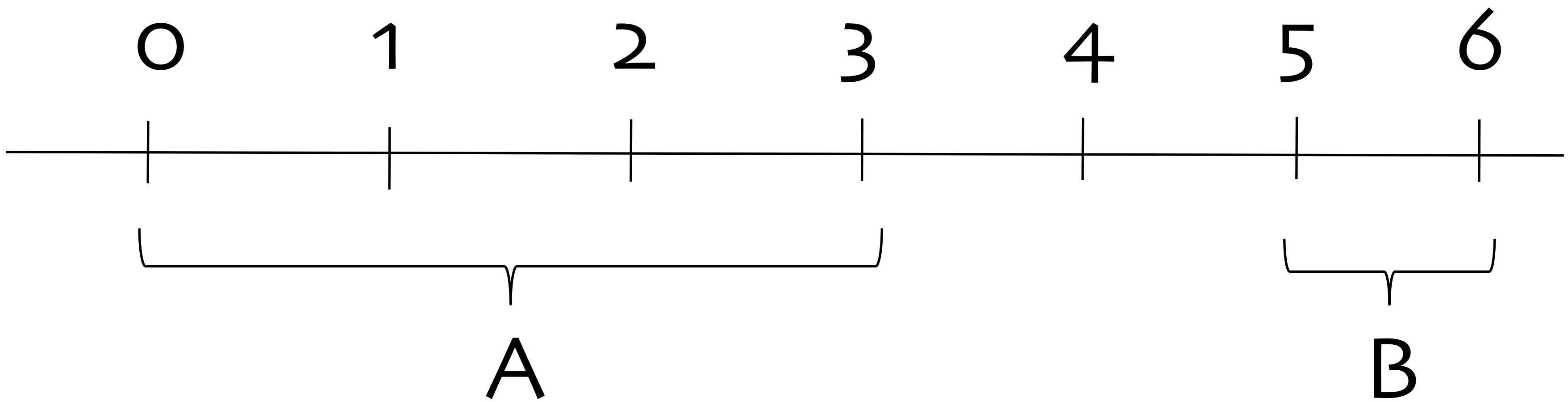


$A = \{0,1,2,3\}$

$B = \{5,6\}$

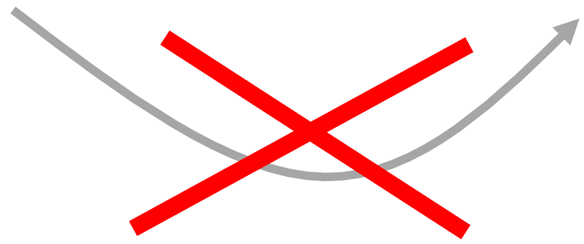
$A \cap B = \emptyset$

$A \cup B = \{3,4,5\} = \{x \mid 3 \leq x \leq 5\}$



WAGNER PARSIMONY

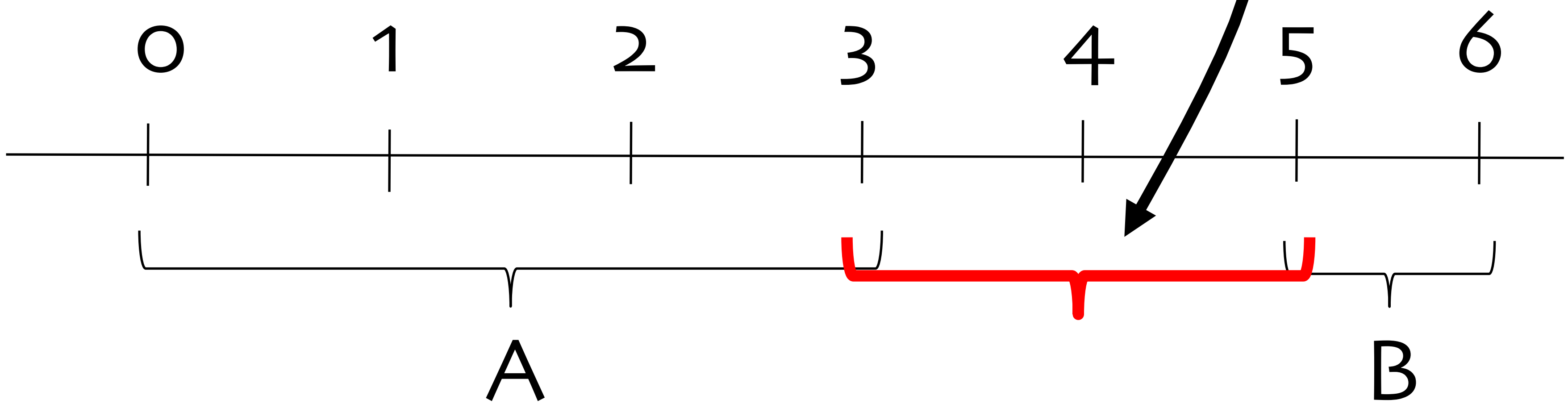
0 <--> 1 <--> 2

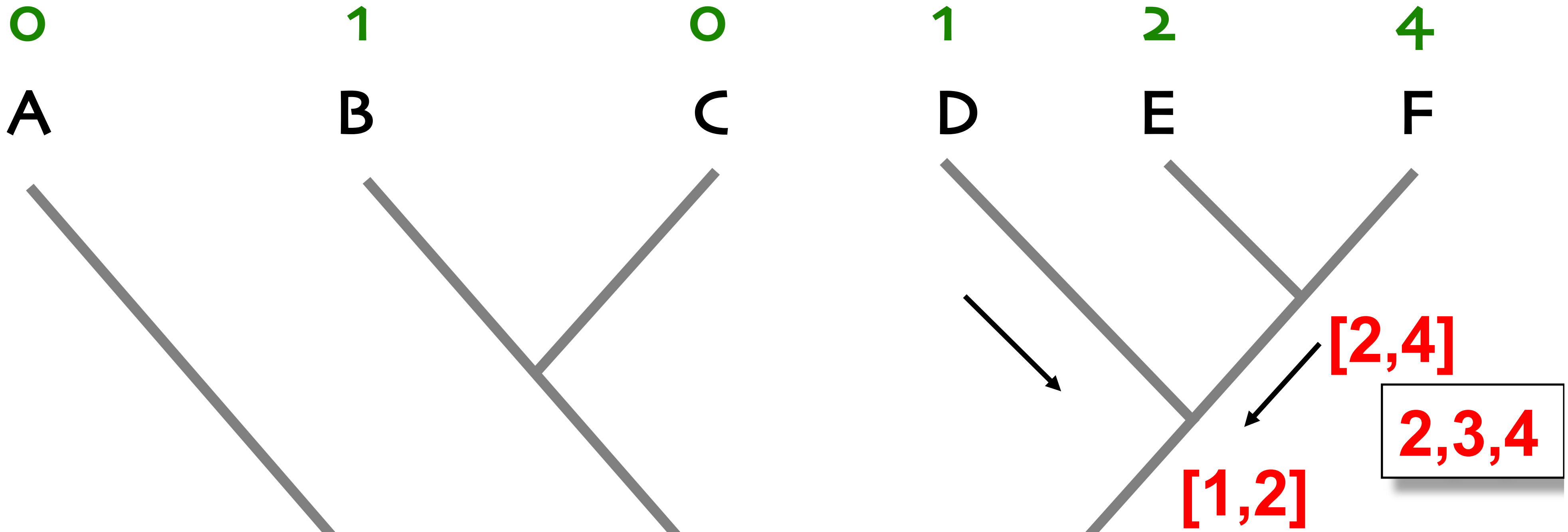


$A = \{0,1,2,3\}$ $\quad A \cap B = \emptyset$

$B = \{5,6\}$ $\quad A \cup B = \{3,4,5\} = \{x \mid 3 \leq x \leq 5\}$

SMALLEST closed interval between A & B



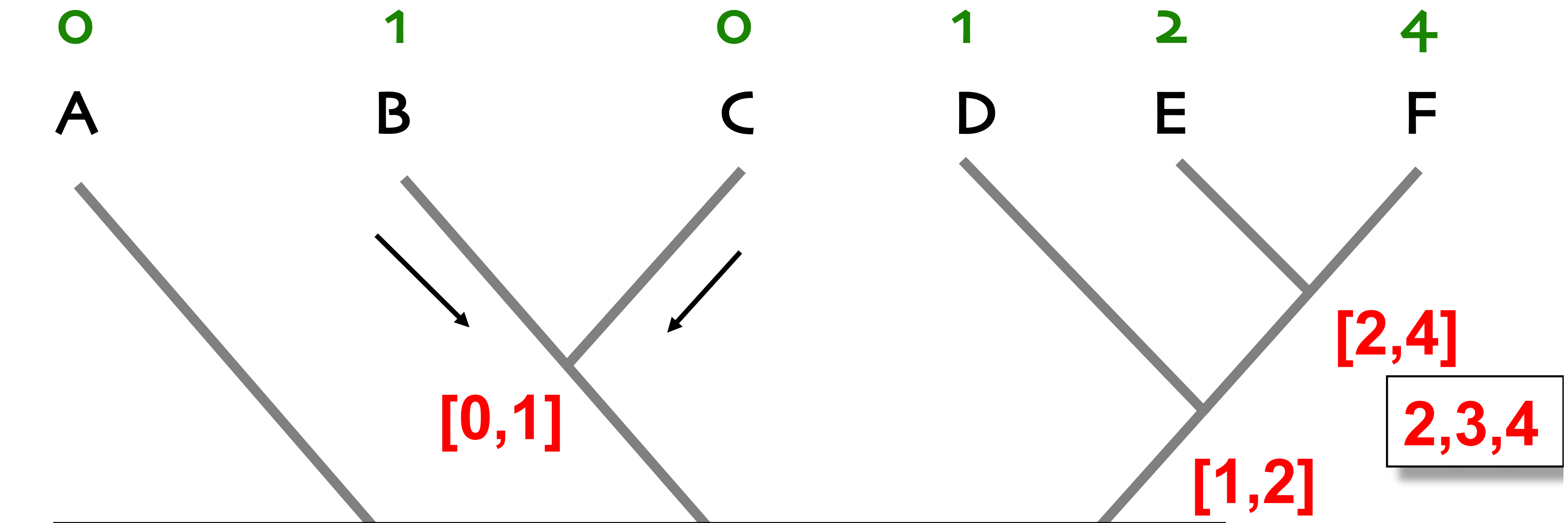


RULE 1: if terminals share character state this will be marked also for their ancestor (intersection, $\cap$)

RULE 2: if terminals do not share ch. states (intersection, $\cap = \emptyset$) assign smallest closed interval between states of terminals for their ancestor, i.e.

$$[a, b] = \{x | a \leq x \leq b\}$$

DOWNWARD PASS



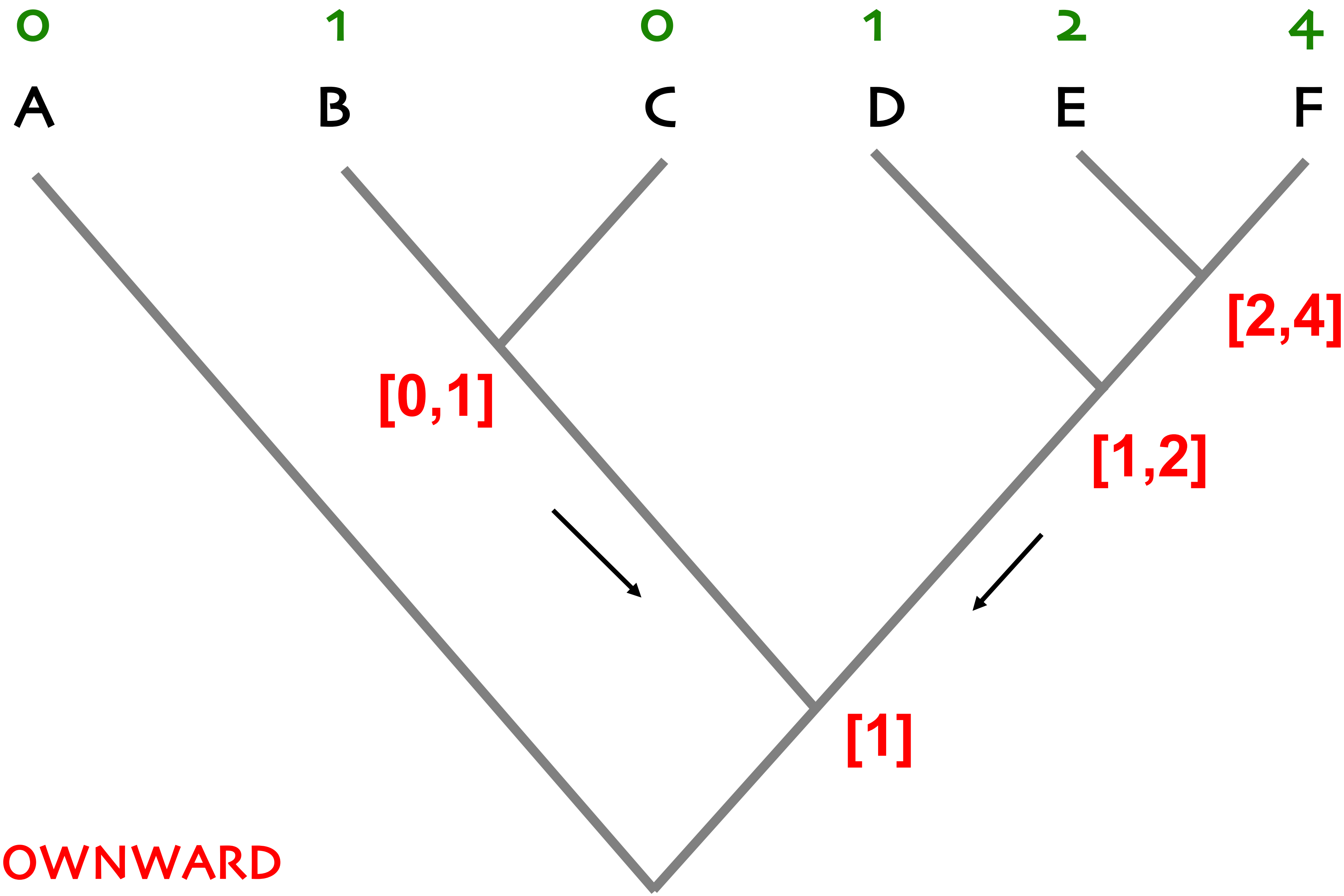
RULE 1: if terminals share character state this will be marked also for their ancestor (intersection, $\cap$)

RULE 2: if terminals do not share ch. states (intersection, $\cap = \emptyset$) assign smallest closed interval between states of terminals for their ancestor, i.e.

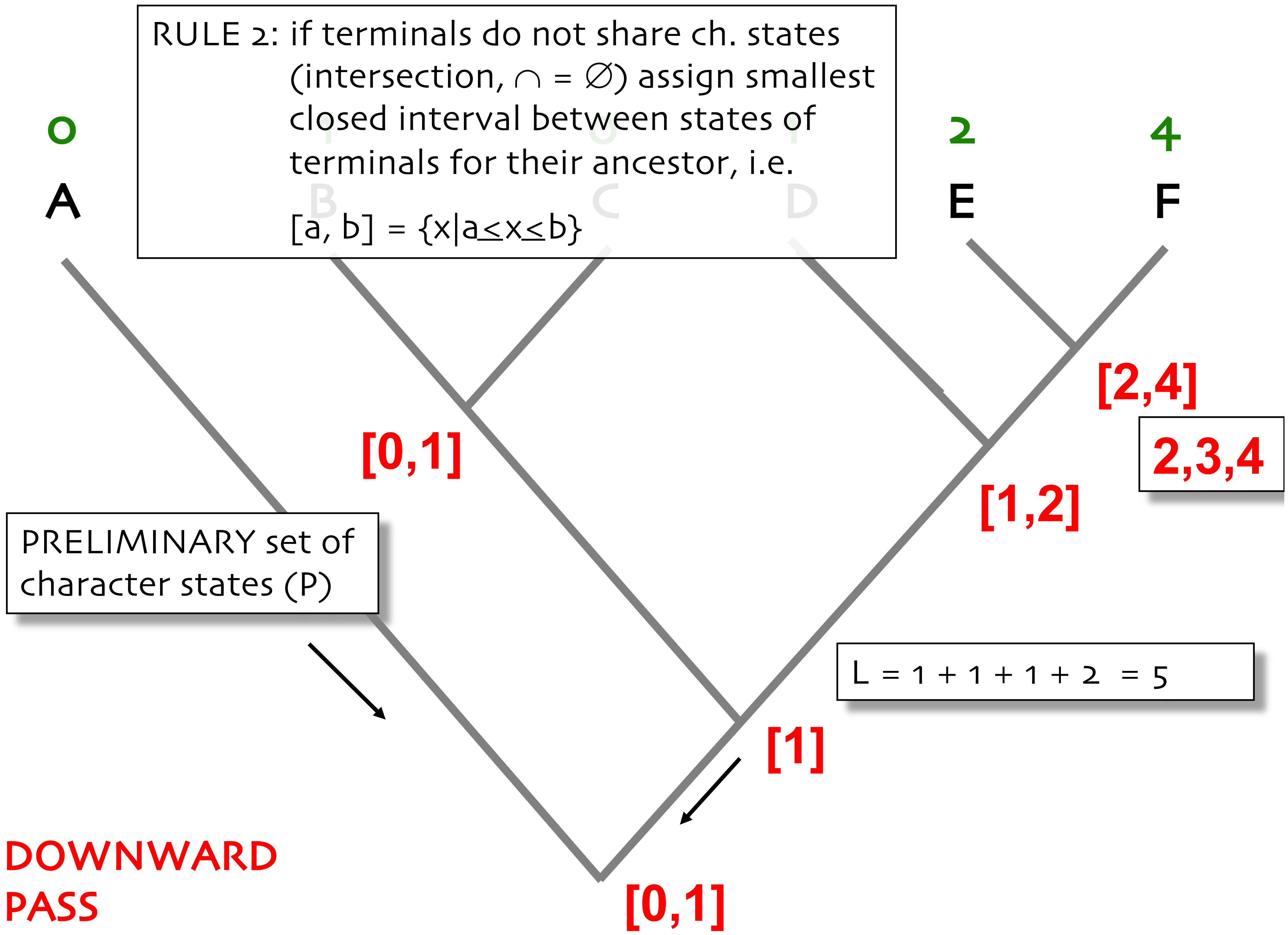
$[a, b] = \{x | a \leq x \leq b\}$

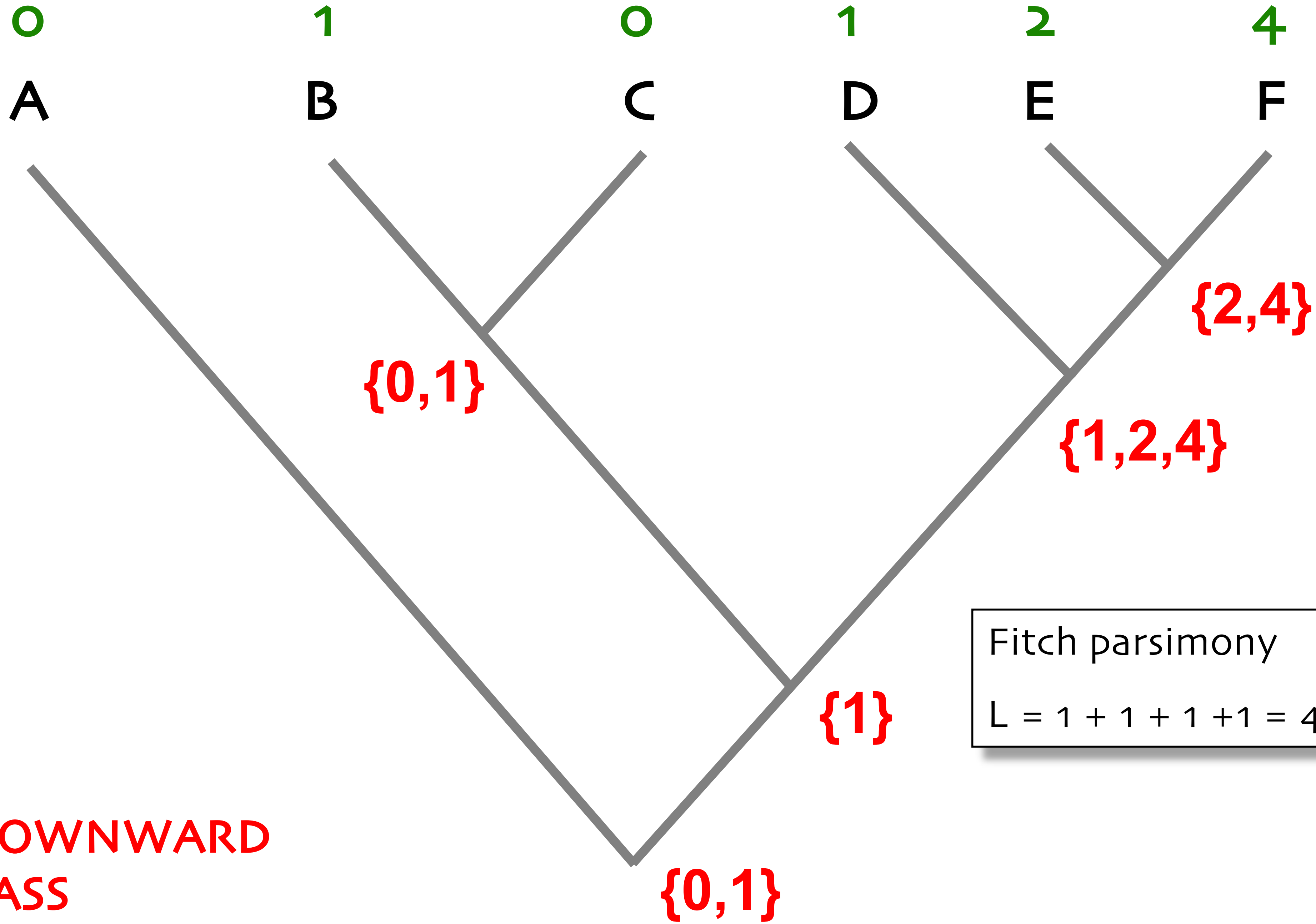
DOWNWARD PASS

RULE 1: if terminals share character state this will be marked also for their ancestor (intersection, $\cap$)



DOWNWARD PASS


RULE 2: if terminals do not share ch. states (intersection, ∩ = ∅) assign smallest closed interval between states of terminals for their ancestor, i.e.
[a, b] = {x|a≤x≤b}
0
A
B
C
D
2
E
4
F
[2,4]
2,3,4
[0,1]
[1,2]
PRELIMINARY set of character states (P)
L = 1 + 1 + 1 + 2 = 5
[1]
[0,1]
DOWNWARD PASS



Fitch parsimony

L = 1 + 1 + 1 +1 = 4

DOWNWARD PASS

" upward pass" rules (Goloboff 1993)

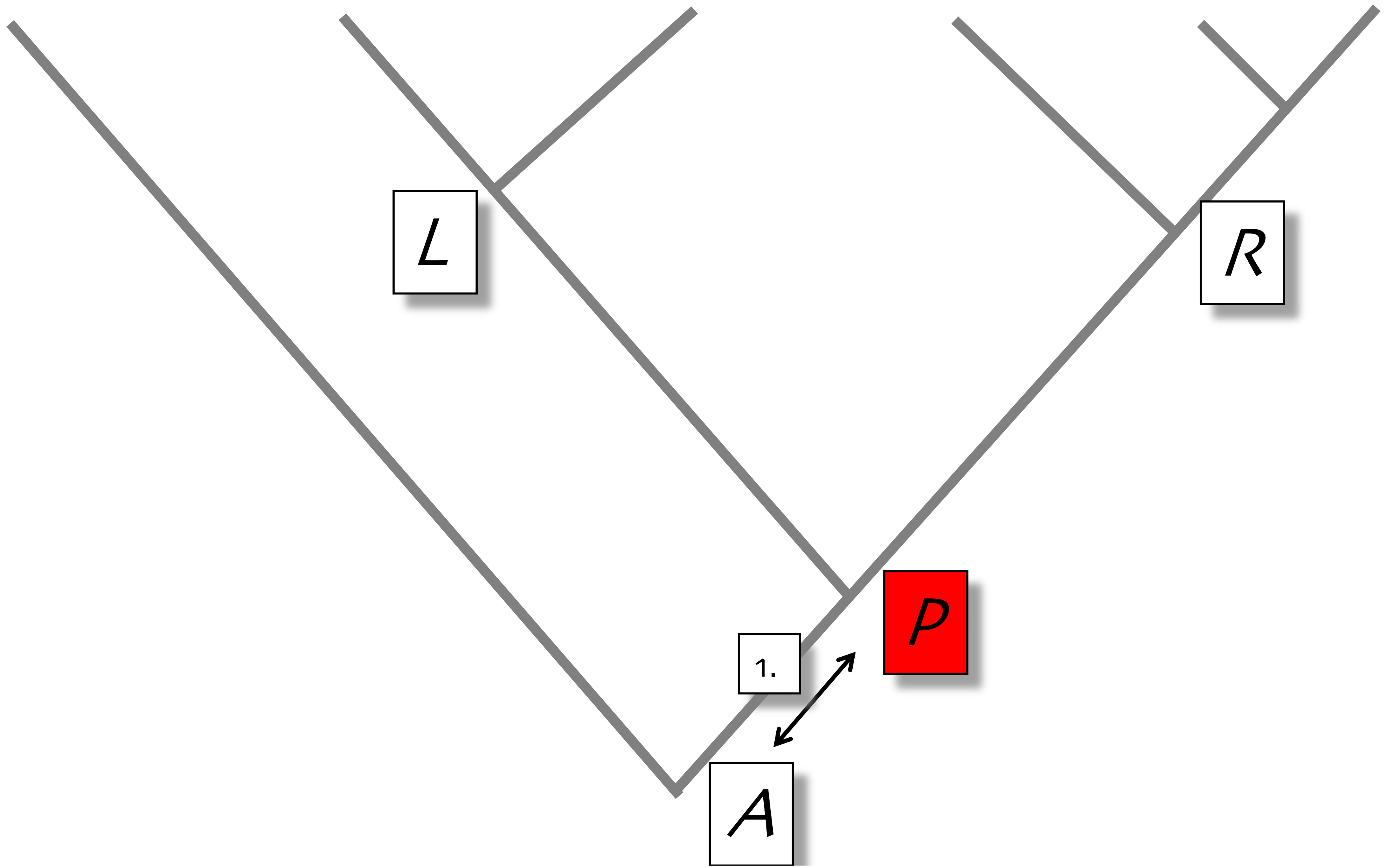
- *PRELIMINARY* (*P*) state set for root and terminals is their final set ($P = F$)

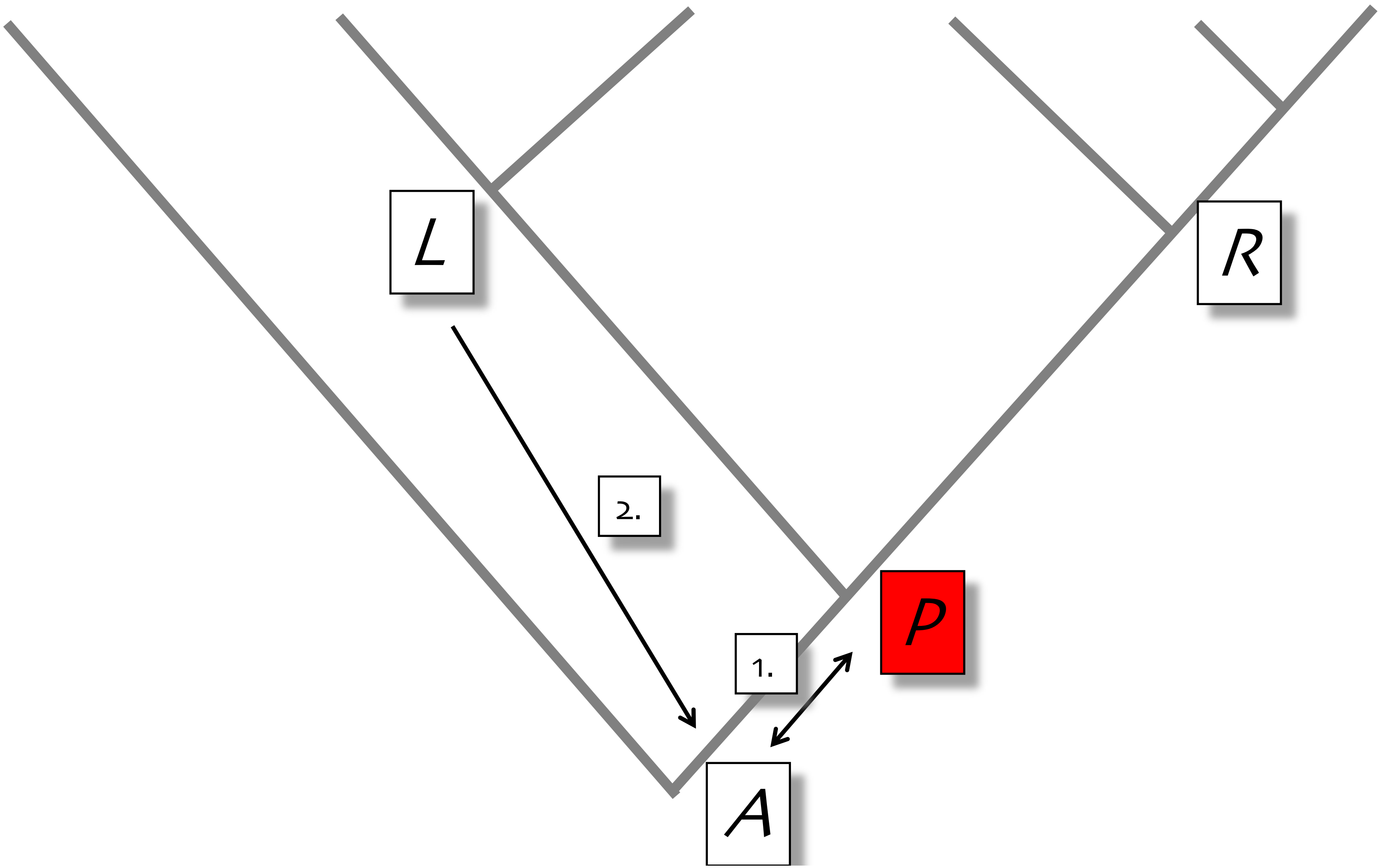
A, character state of immediate ancestor

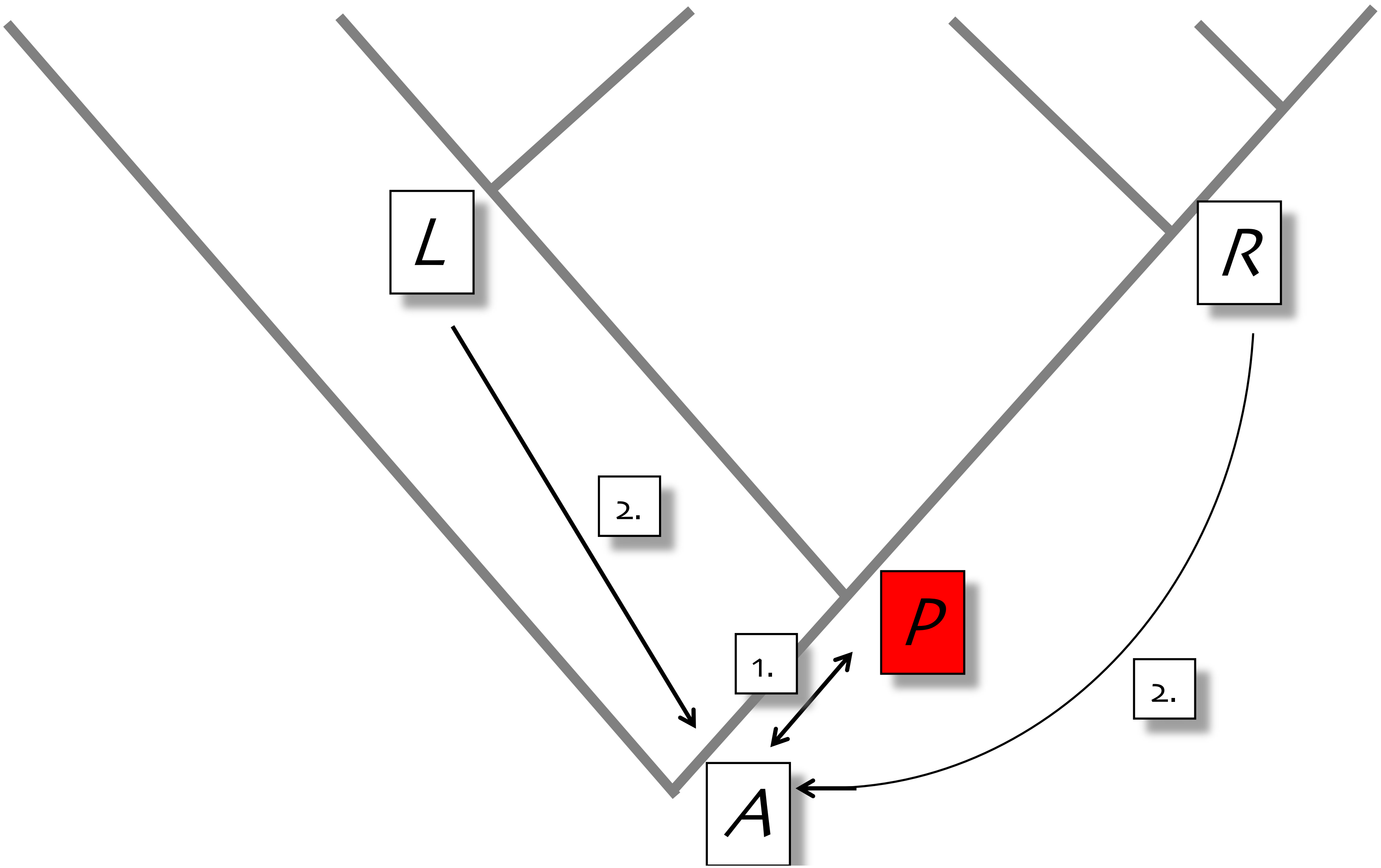
RULE 1. If $A \cap P = A$, $F = A$.

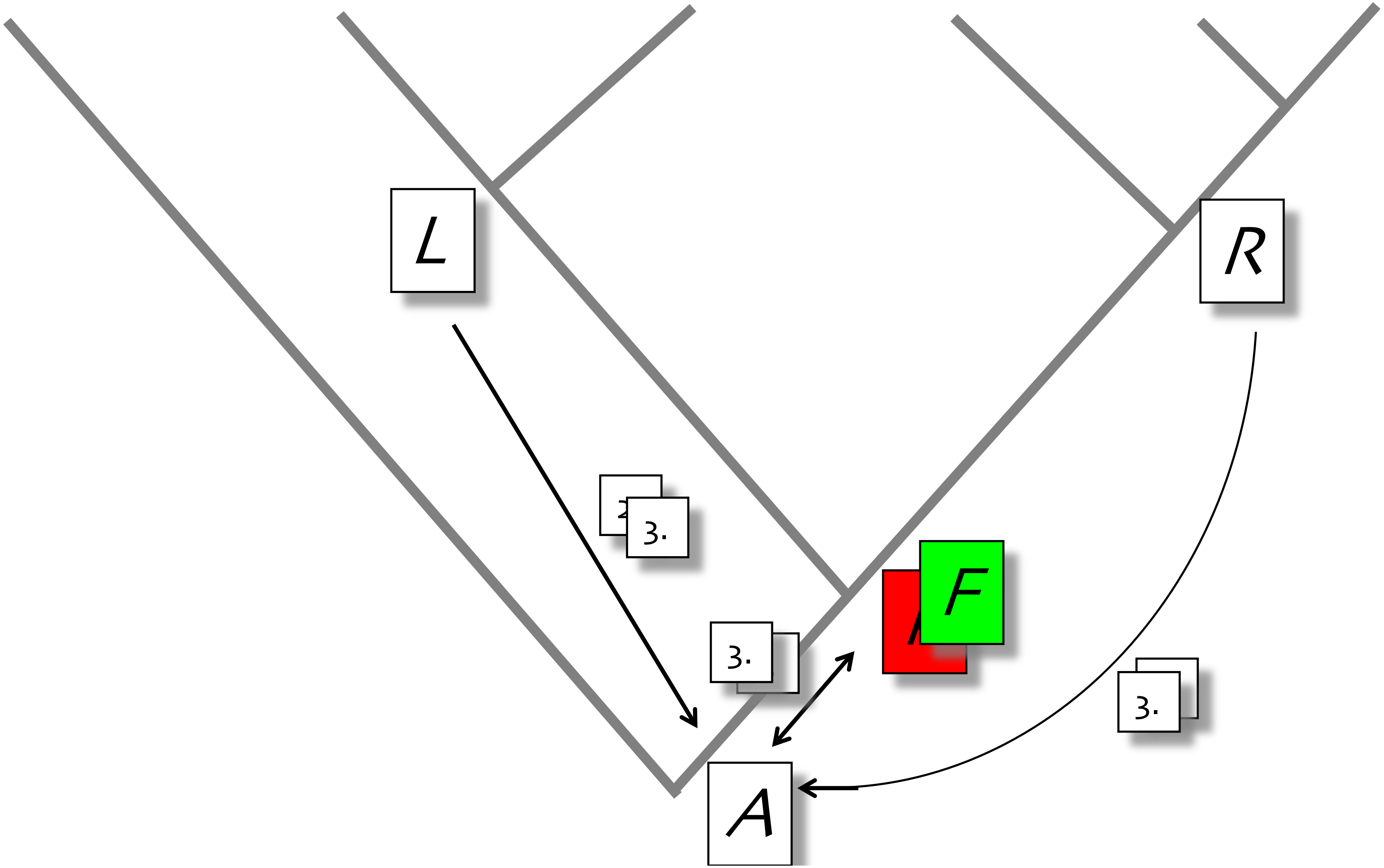
RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \emptyset$, define X as $X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \emptyset$, $F = X$. If $X \cap P = \emptyset$, F equals the LARGEST closed interval between X and state in P closest to X.

RULE 3. If rules 1 & 2 do not apply, F equals the LARGEST closed interval between the state in P closest to A and the state in $(L \cup R)$ closest to A.


L
R
P
1.
A


L
R
2.
P
1.
A


L
R
2.
P
1.
2.
A



L
R
3.
3.
3.
F
A

" upward pass" rules (Goloboff 1993)

- *PRELIMINARY* (*P*) state set for root and terminals is their final set ($P = F$)

A, character state of immediate ancestor

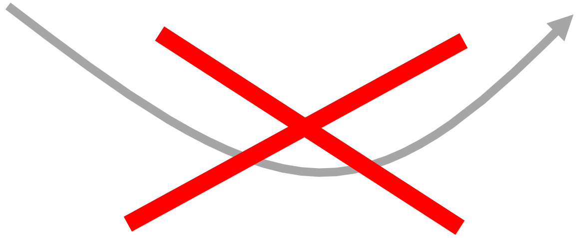
RULE 1. If $A \cap P = A$, $F = A$.

RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \emptyset$, define X as $X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \emptyset$, $F = X$. If $X \cap P = \emptyset$, F **equals the LARGEST closed interval between *X* and state in *P* closest to *X*.**

RULE 3. If rules 1 & 2 do not apply, *F* equals the LARGEST closed interval between the state in *P* closest to *A* and the state in $(L \cup R)$ closest to A.

WAGNER PARSIMONY

0 <--> 1 <--> 2



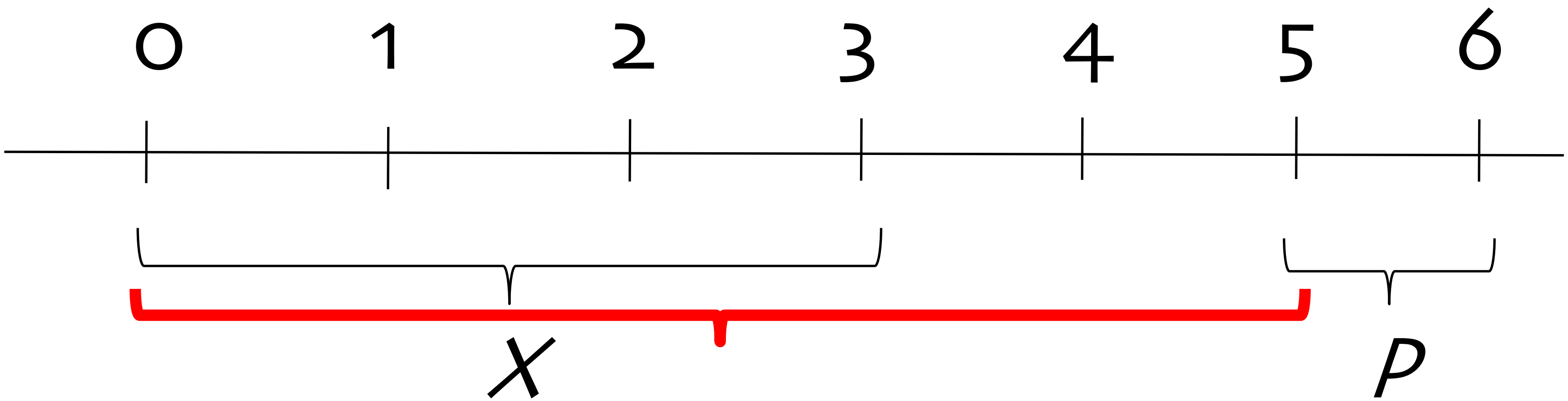
RULE 2.

$X = \{0,1,2,3\}$

$P = \{5,6\}$

$X \cap P = \emptyset$

$F = [\mathbf{0,5}] = \{x | 0 \leq x \leq \mathbf{5}\}$



" upward pass" rules (Goloboff 1993)

- *PRELIMINARY* (*P*) state set for root and terminals is their final set ($P = F$)

A, character state of immediate ancestor

RULE 1. If $A \cap P = A$, $F = A$.

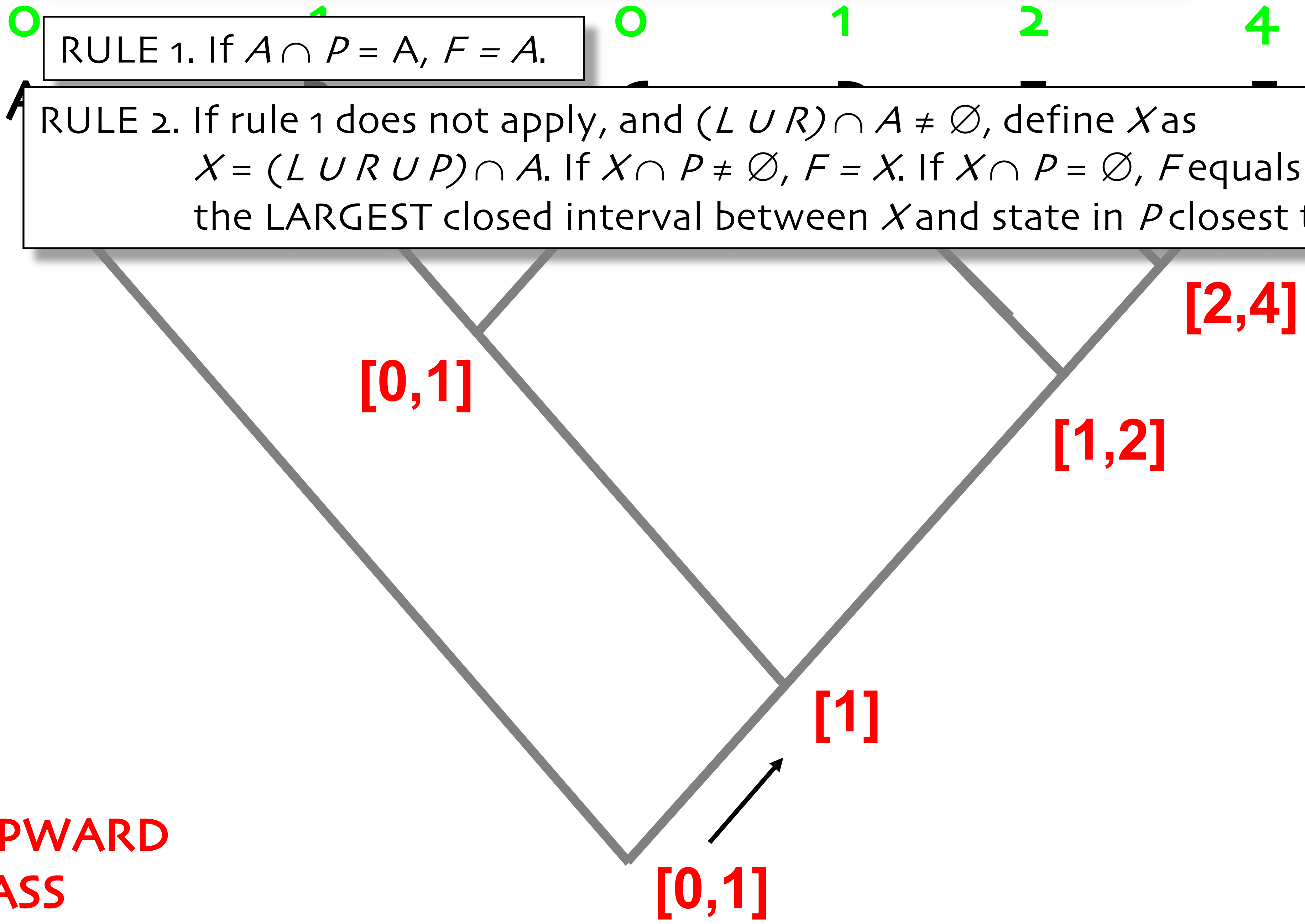
RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \emptyset$, define *X* as $X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \emptyset$, $F = X$. If $X \cap P = \emptyset$, *F* equals the LARGEST closed interval between *X* and state in *P* closest to *X*.

RULE 3. If rules 1 & 2 do not apply, *F* equals the LARGEST closed interval between the state in *P* closest to *A* and the state in $(L \cup R)$ closest to A.

PRELIMINARY (P) state set for root and terminals is their final set ($P = F$)

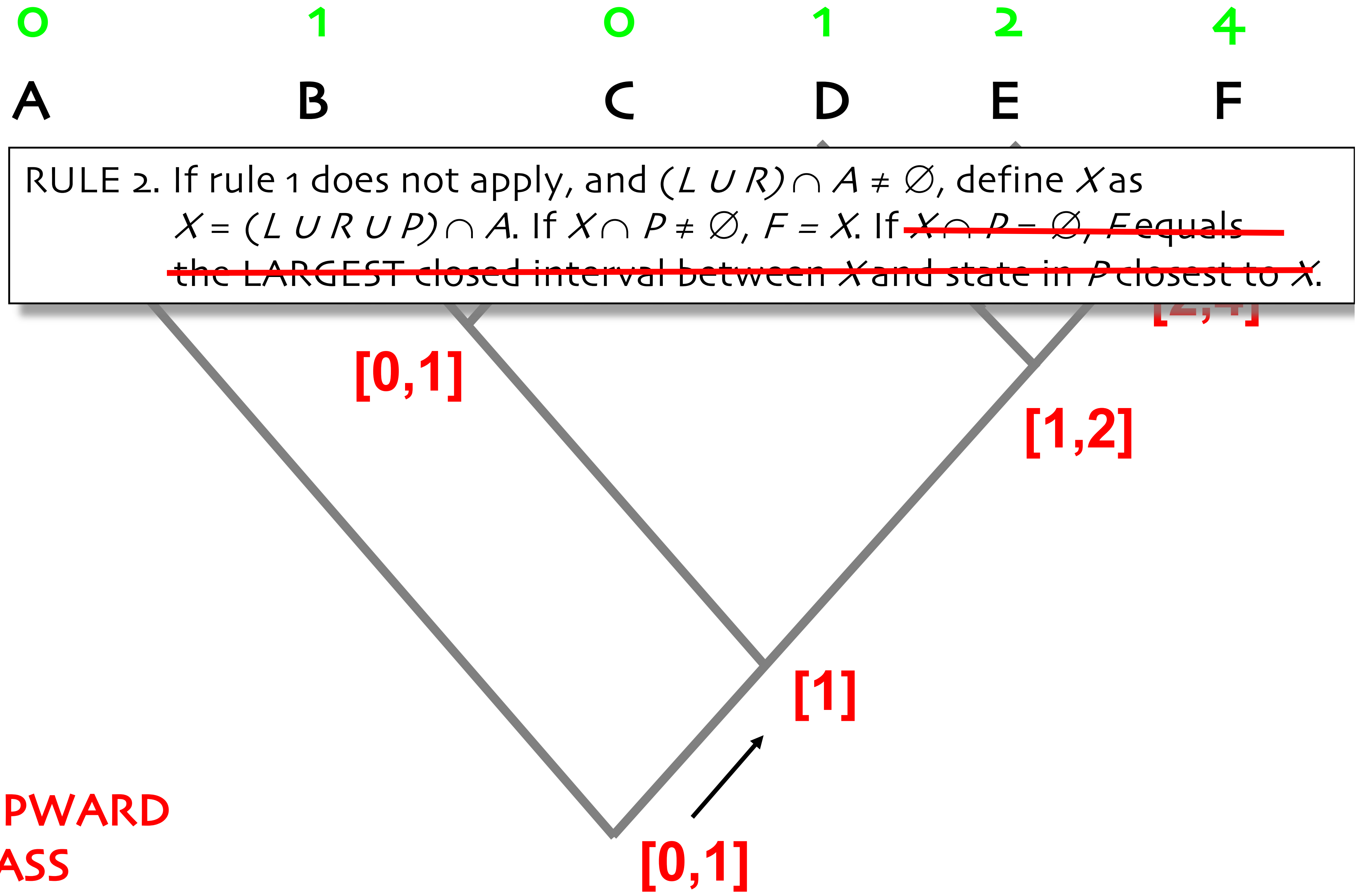
RULE 1. If $A \cap P = A$, $F = A$.

RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \emptyset$, define X as $X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \emptyset$, $F = X$. If $X \cap P = \emptyset$, F equals the LARGEST closed interval between X and state in P closest to X.

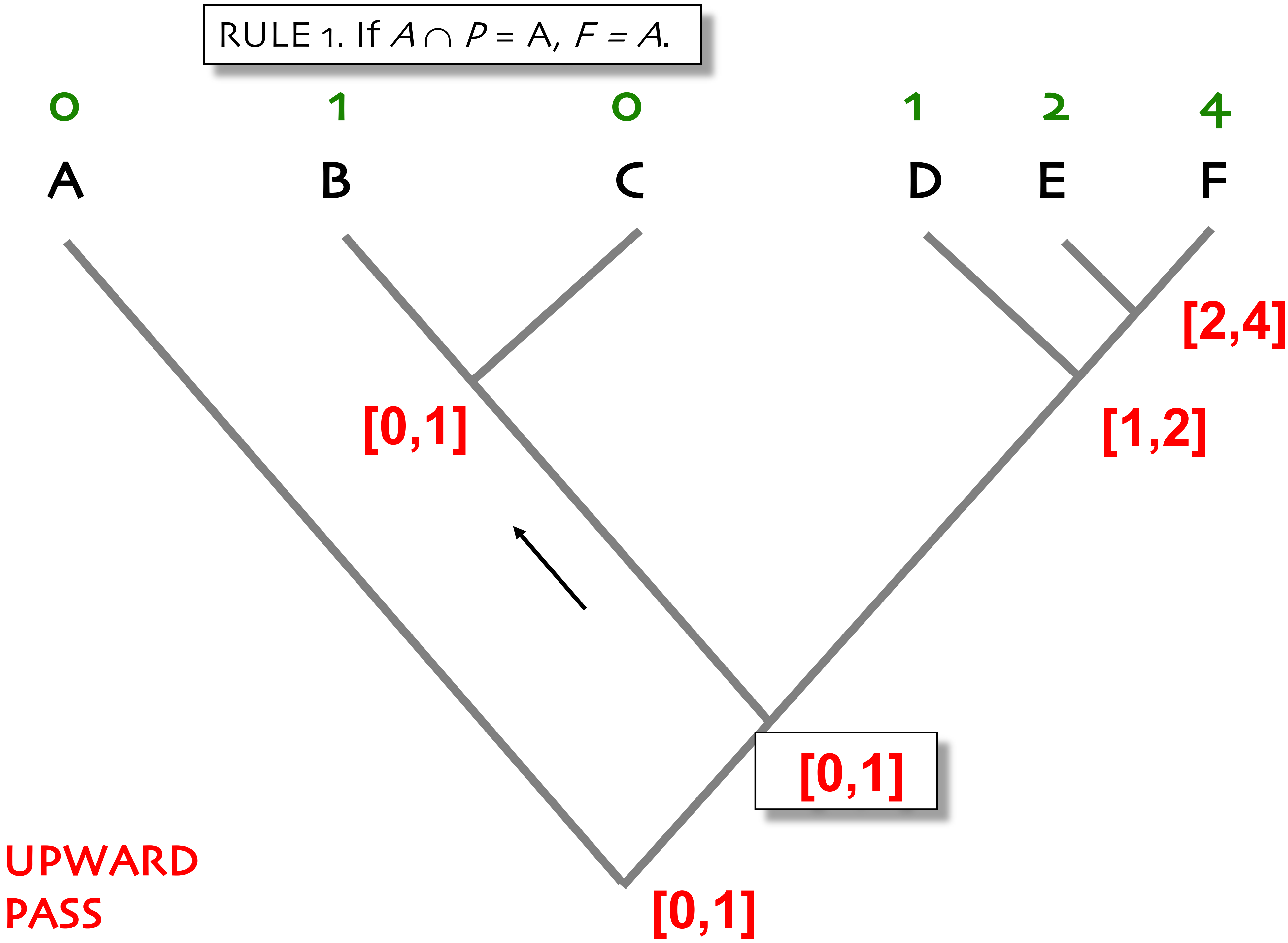


UPWARD PASS

RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \varnothing$, define X as $X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \varnothing$, $F = X$. If ~~$X \cap P = \varnothing$, F equals the LARGEST closed interval between X and state in P closest to X.~~



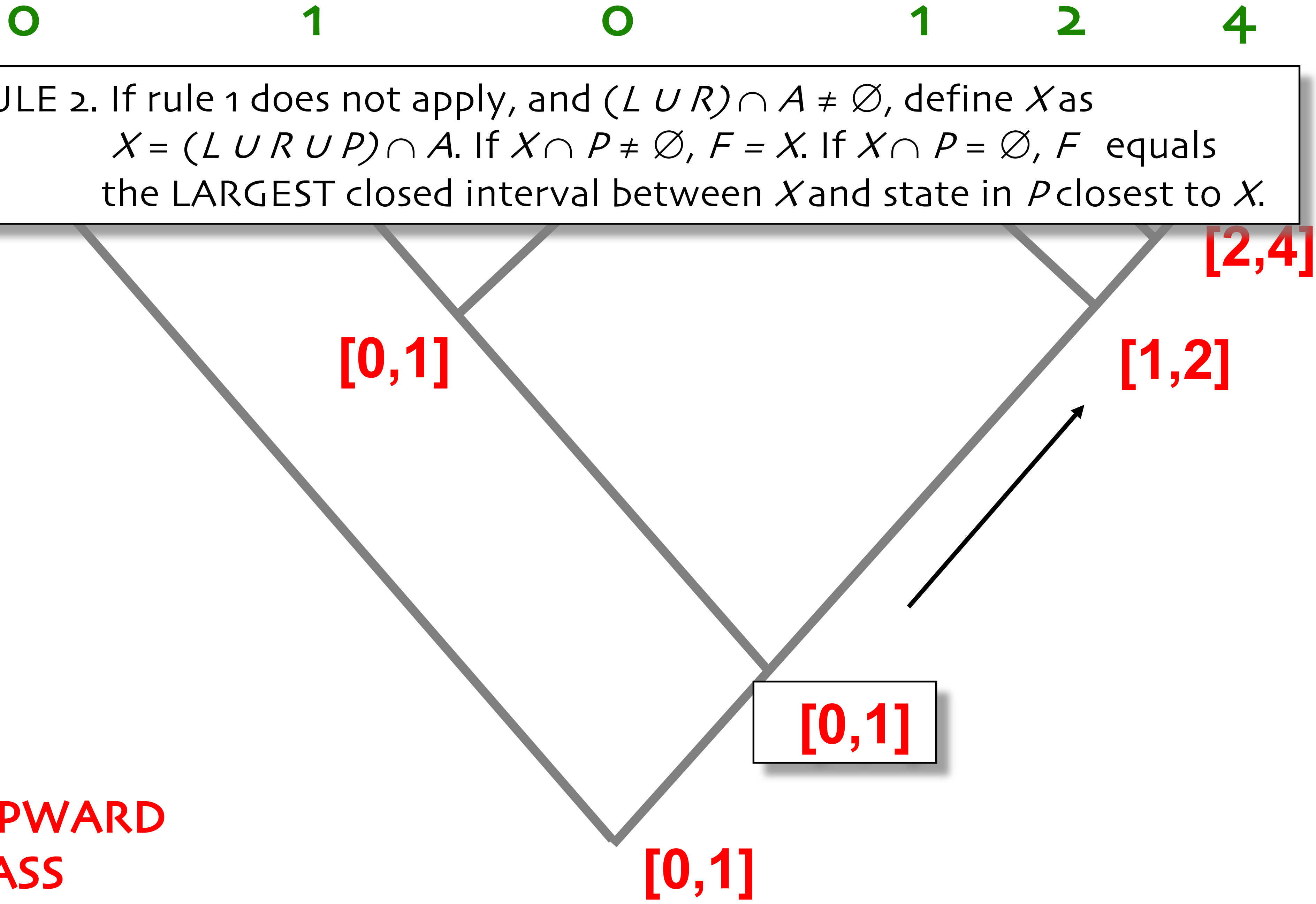
UPWARD PASS



RULE 1. If $A \cap P = A$, $F = A$.
0
1
0
1
2
4
A
B
C
D
E
F
[2,4]
[0,1]
[1,2]
[0,1]
[0,1]
UPWARD PASS

RULE 1. If $A \cap P = A$, $F = A$.

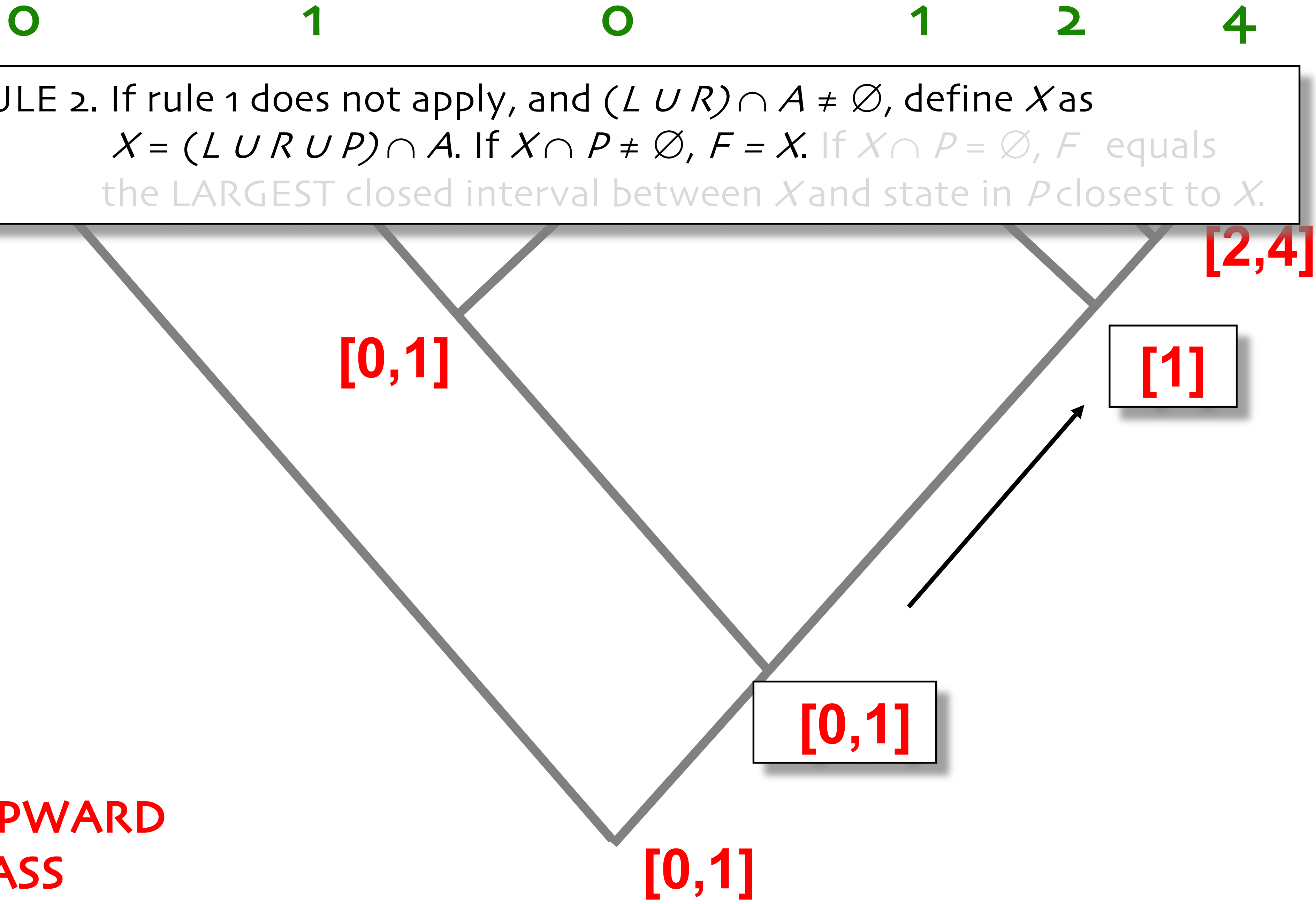
RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \emptyset$, define X as $X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \emptyset$, $F = X$. If $X \cap P = \emptyset$, F equals the LARGEST closed interval between X and state in P closest to X.



UPWARD PASS

RULE 1. If $A \cap P$ = A, $F = A$.

RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \emptyset$, define X as
$X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \emptyset$, $F = X$. If $X \cap P = \emptyset$, F equals the LARGEST closed interval between X and state in P closest to X.

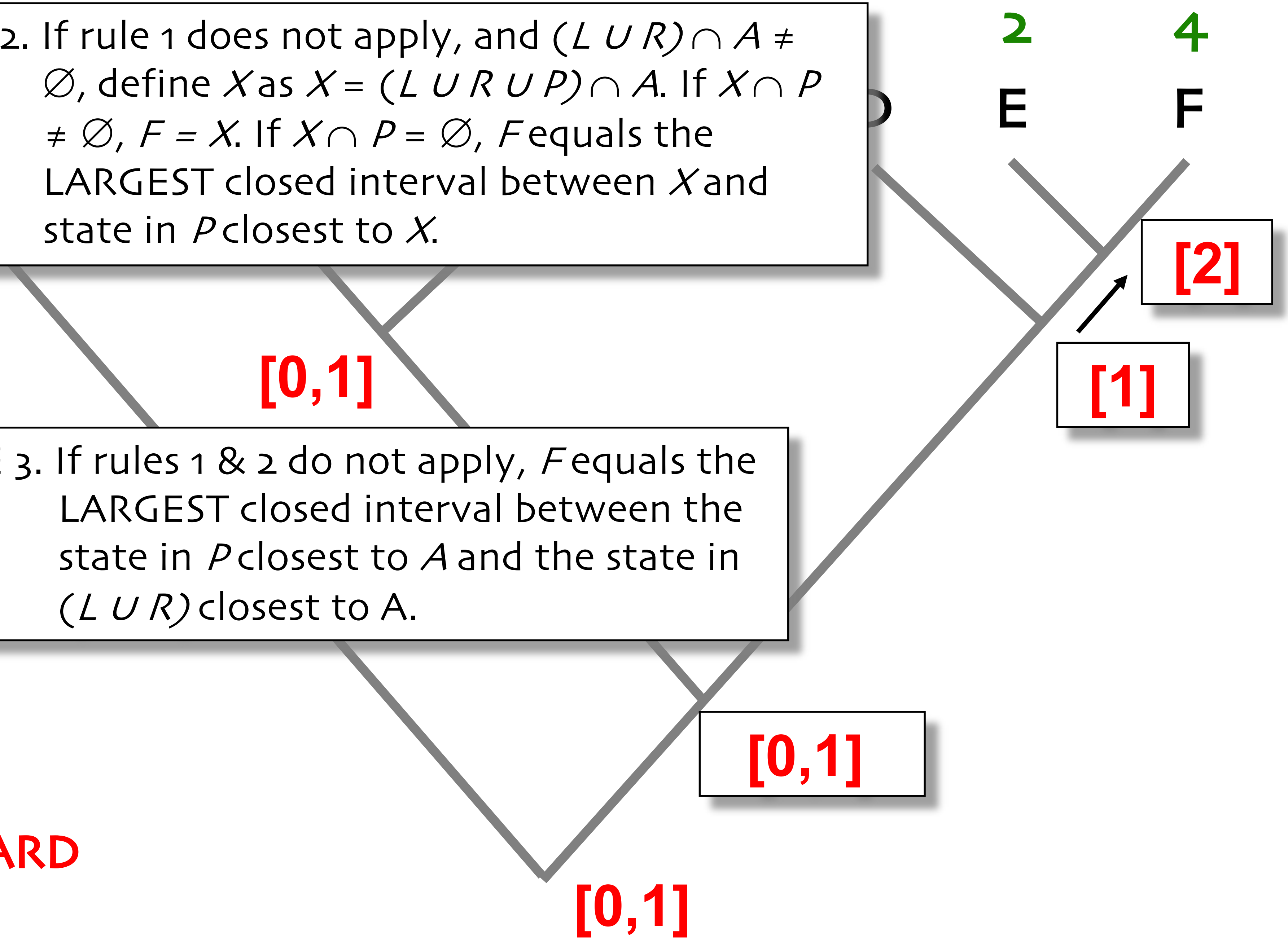


UPWARD
PASS

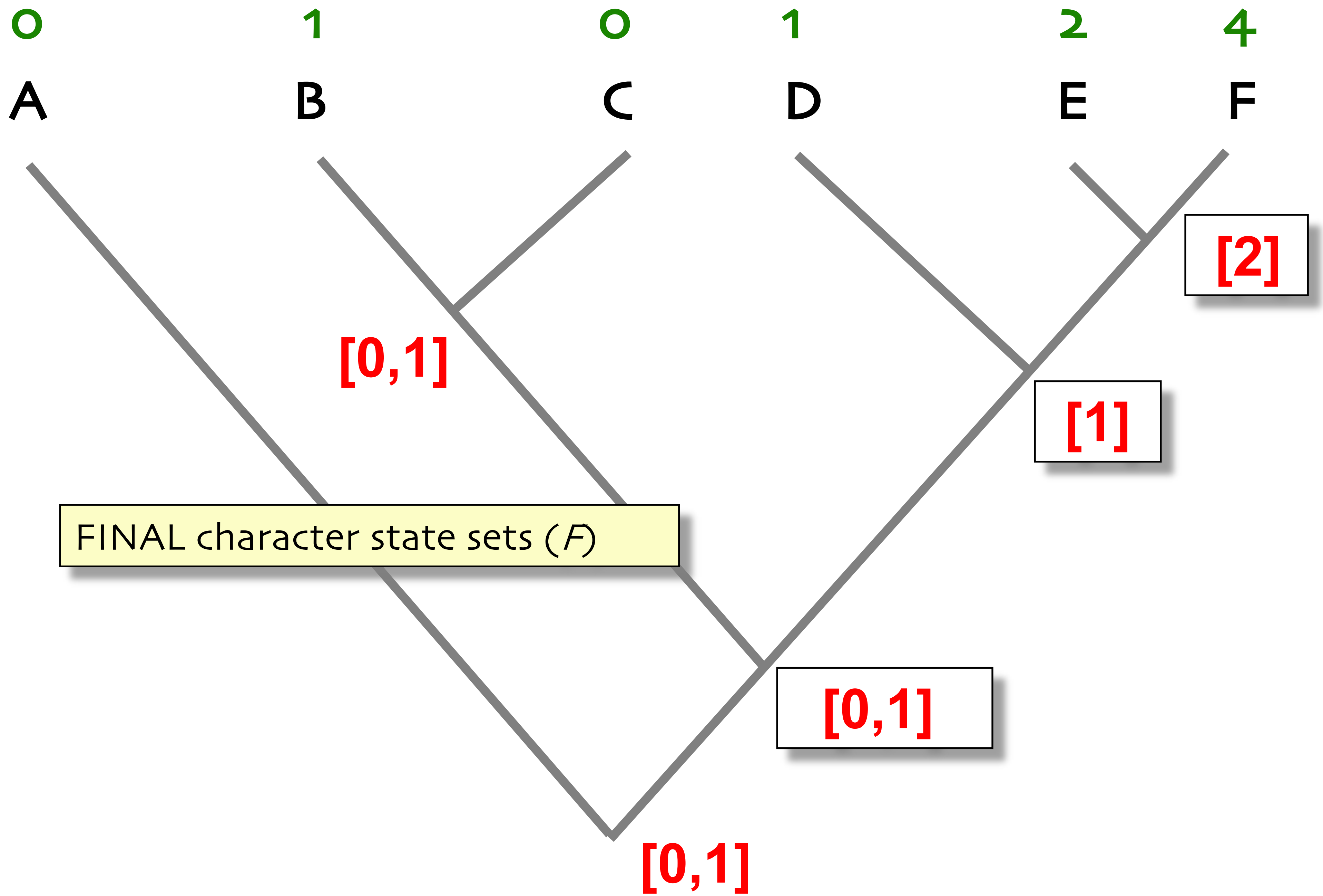
RULE 1. If $A \cap P$ = A, $F = A$.

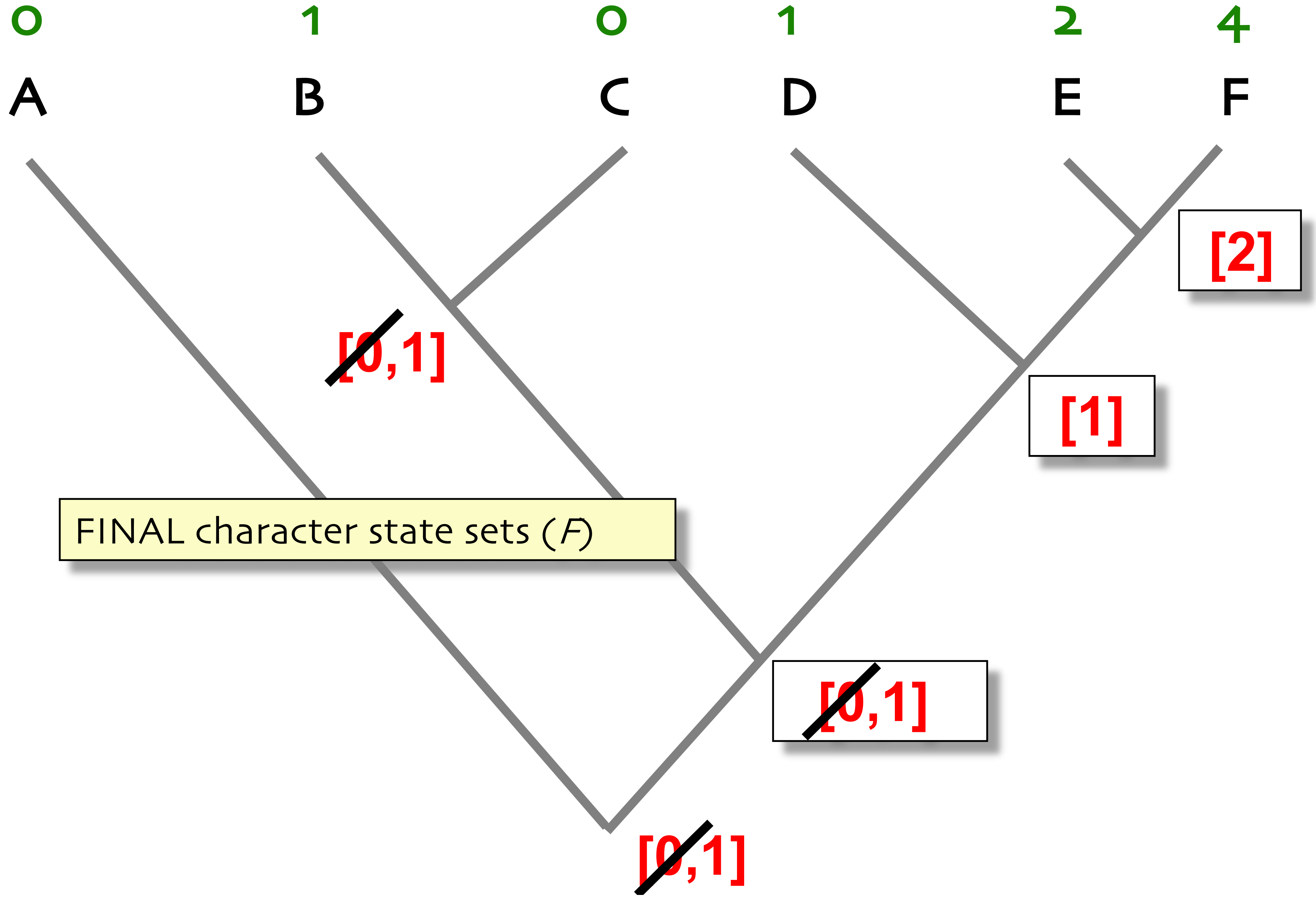
RULE 2. If rule 1 does not apply, and $(L \cup R) \cap A \neq \varnothing$, define X as $X = (L \cup R \cup P) \cap A$. If $X \cap P \neq \varnothing$, $F = X$. If $X \cap P = \varnothing$, F equals the LARGEST closed interval between X and state in P closest to X.

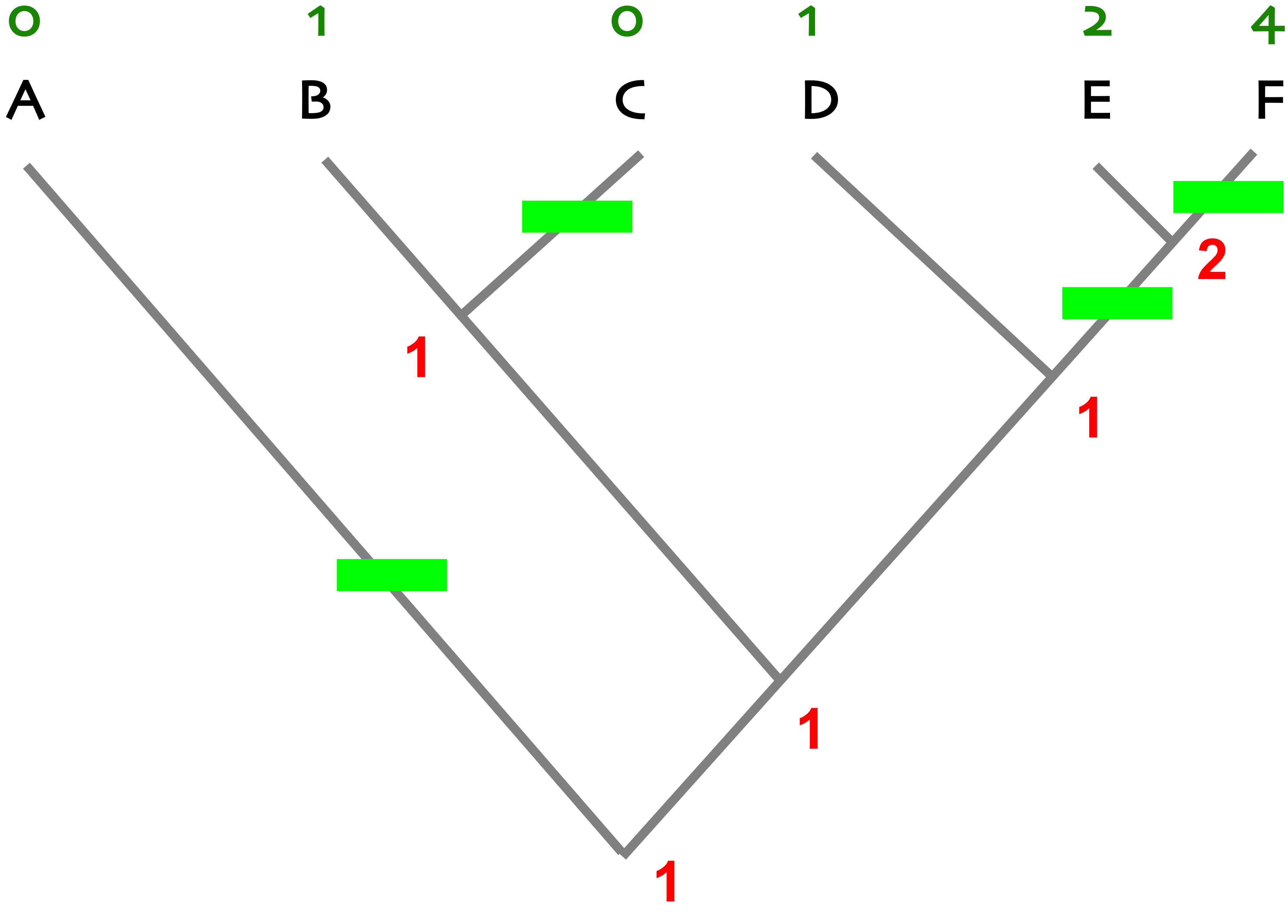
RULE 3. If rules 1 & 2 do not apply, F equals the LARGEST closed interval between the state in P closest to A and the state in $(L \cup R)$ closest to A.

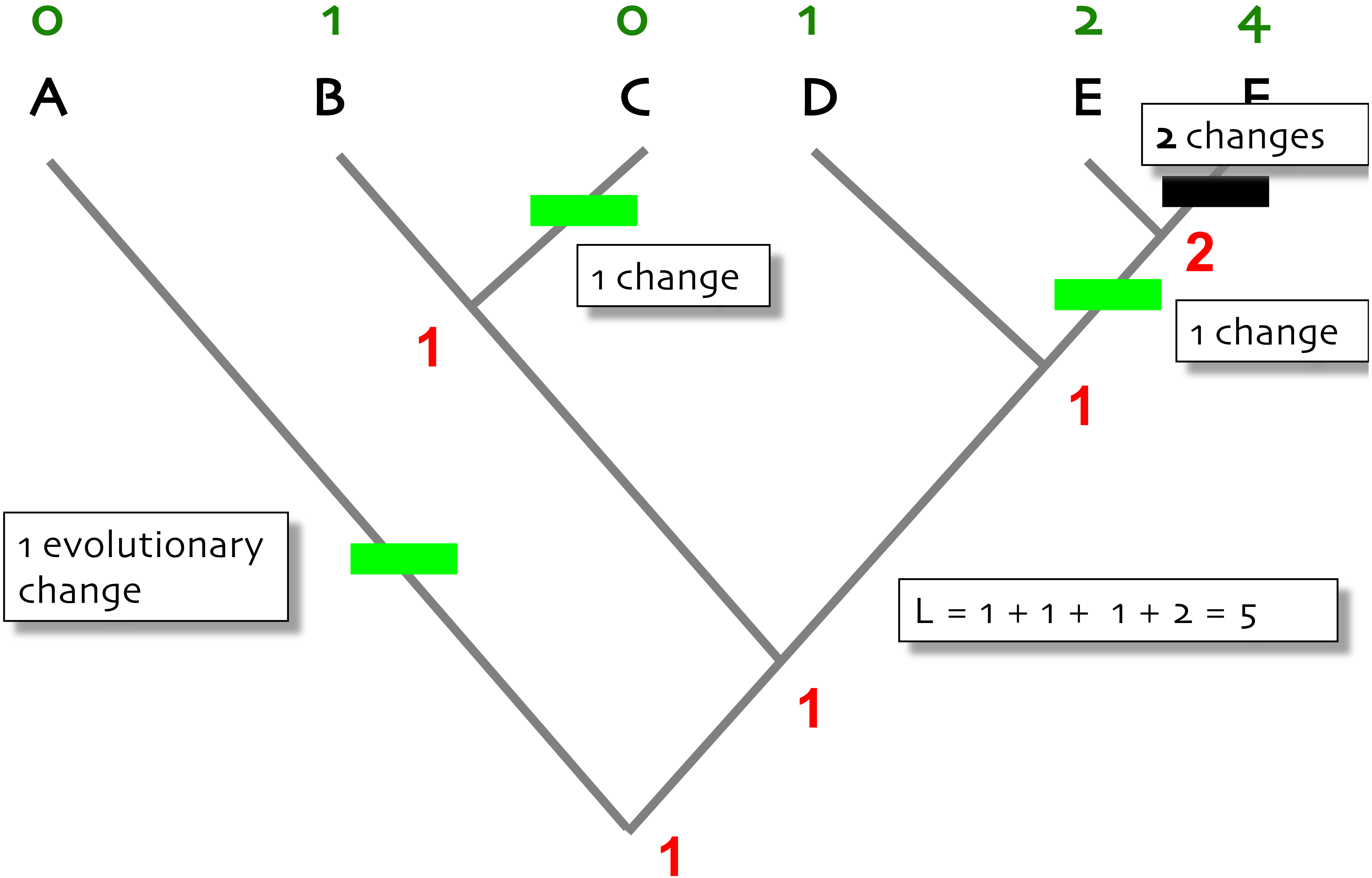


UPWARD PASS

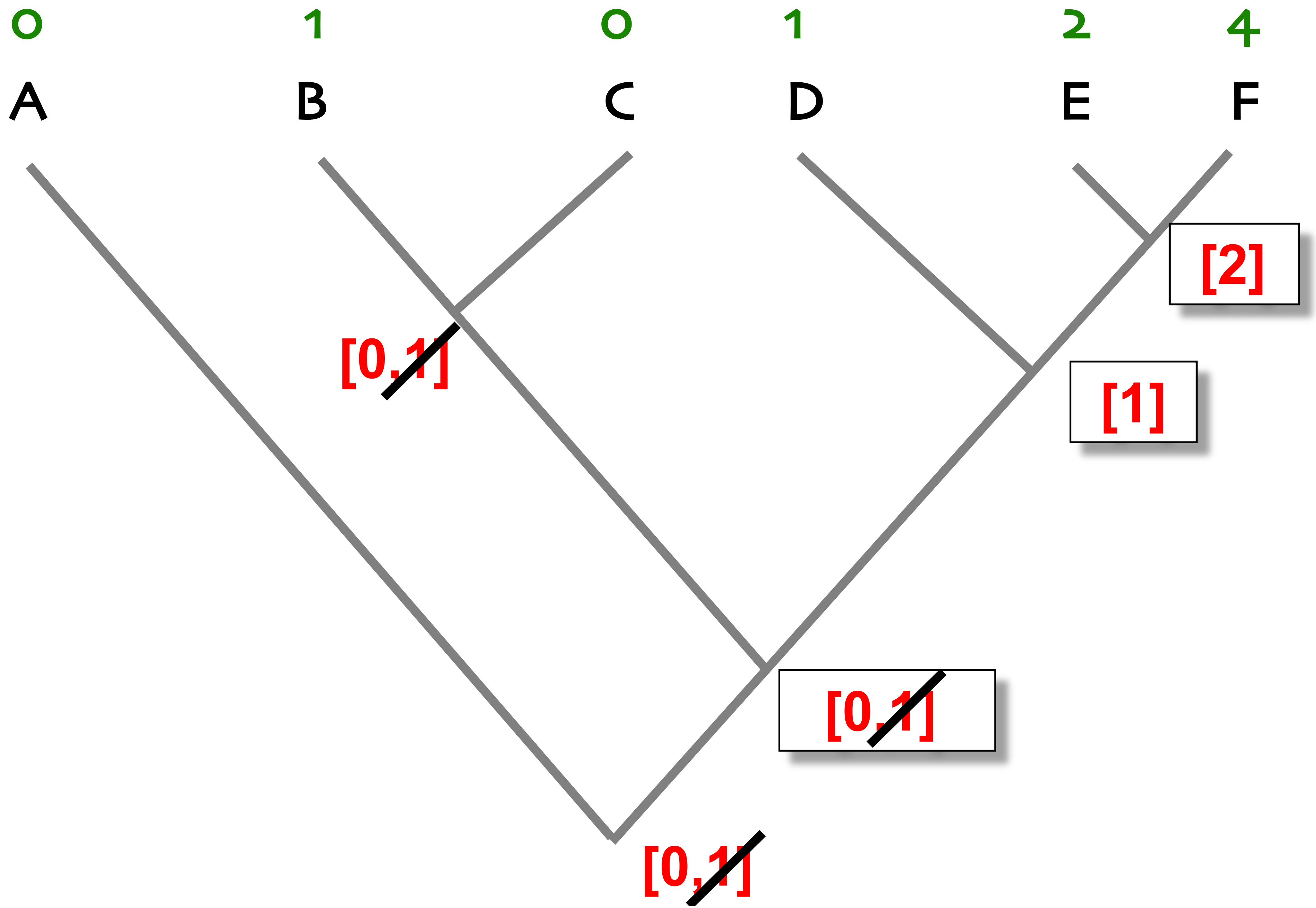

0
1
0
1
2
4
A
B
C
D
E
F
[2]
[0,1]
[1]
FINAL character state sets (F)
[0,1]
[0,1]


0
1
0
1
2
4
A
B
C
D
E
F
[2]
[0,1]
[1]
FINAL character state sets (F)
[0,1]
[0,1]

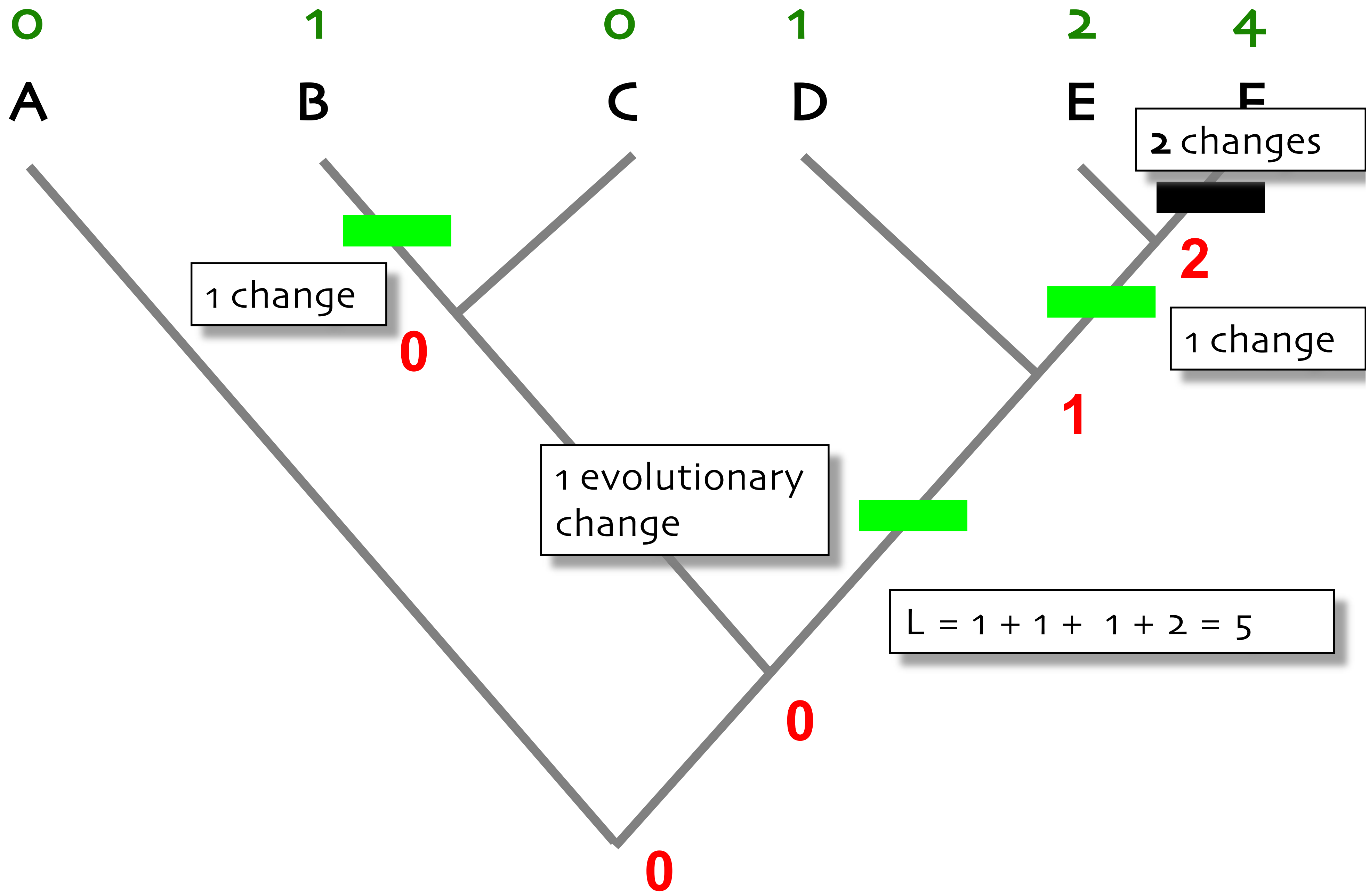

0
1
0
1
2
4
A
B
C
D
E
F
1
2
1
1
1



0
1
0
1
2
4
A
B
C
D
E
F
2 changes
1 change
1 change
2
1
1
1
1
1 evolutionary change
L = 1 + 1 + 1 + 2 = 5



0
1
0
1
2
4
A
B
C
D
E
F
[2]
[0,1]
[1]
[0,1]
[0,1]


0
1
0
1
2
4
A
B
C
D
E
2 changes
1 change
0
2
1 change
1
1 evolutionary change
0
L = 1 + 1 + 1 + 2 = 5
0

SUMMARY

a posteriori character weighting is objective but justification still debated

implied weighting made *during* the search

optimization has to be used in order to find shortest tree & to find character states for internal nodes

MULTIPLE equally parsimonious reconstructions are possible