Semantic Analysis with Attribute Grammars Part 2

Y.N. Srikant

Department of Computer Science and Automation
Indian Institute of Science
Bangalore 560 012

NPTEL Course on Principles of Compiler Design

Outline of the Lecture

- Introduction (covered in lecture 1)
- Attribute grammars
- Attributed translation grammars
- Semantic analysis with attributed translation grammars

Attribute Grammars

- Let $G = (N, T, P, S)$ be a CFG and let $V = N \cup T$.
- Every symbol X of V has associated with it a set of *attributes*
- Two types of attributes: *inherited* and *synthesized*
- Each attribute takes values from a specified domain
- A production $p \in P$ has a set of attribute computation rules for
  - synthesized attributes of the LHS non-terminal of p
  - inherited attributes of the RHS non-terminals of p
- Rules are strictly local to the production p (no side effects)

Synthesized and Inherited Attributes

- An attribute cannot be both synthesized and inherited, but a symbol can have both types of attributes
- Attributes of symbols are evaluated over a parse tree by making passes over the parse tree
- Synthesized attributes are computed in a bottom-up fashion from the leaves upwards
  - Always synthesized from the attribute values of the children of the node
  - Leaf nodes (terminals) have synthesized attributes (only) initialized by the lexical analyzer and cannot be modified
- Inherited attributes flow down from the parent or siblings to the node in question

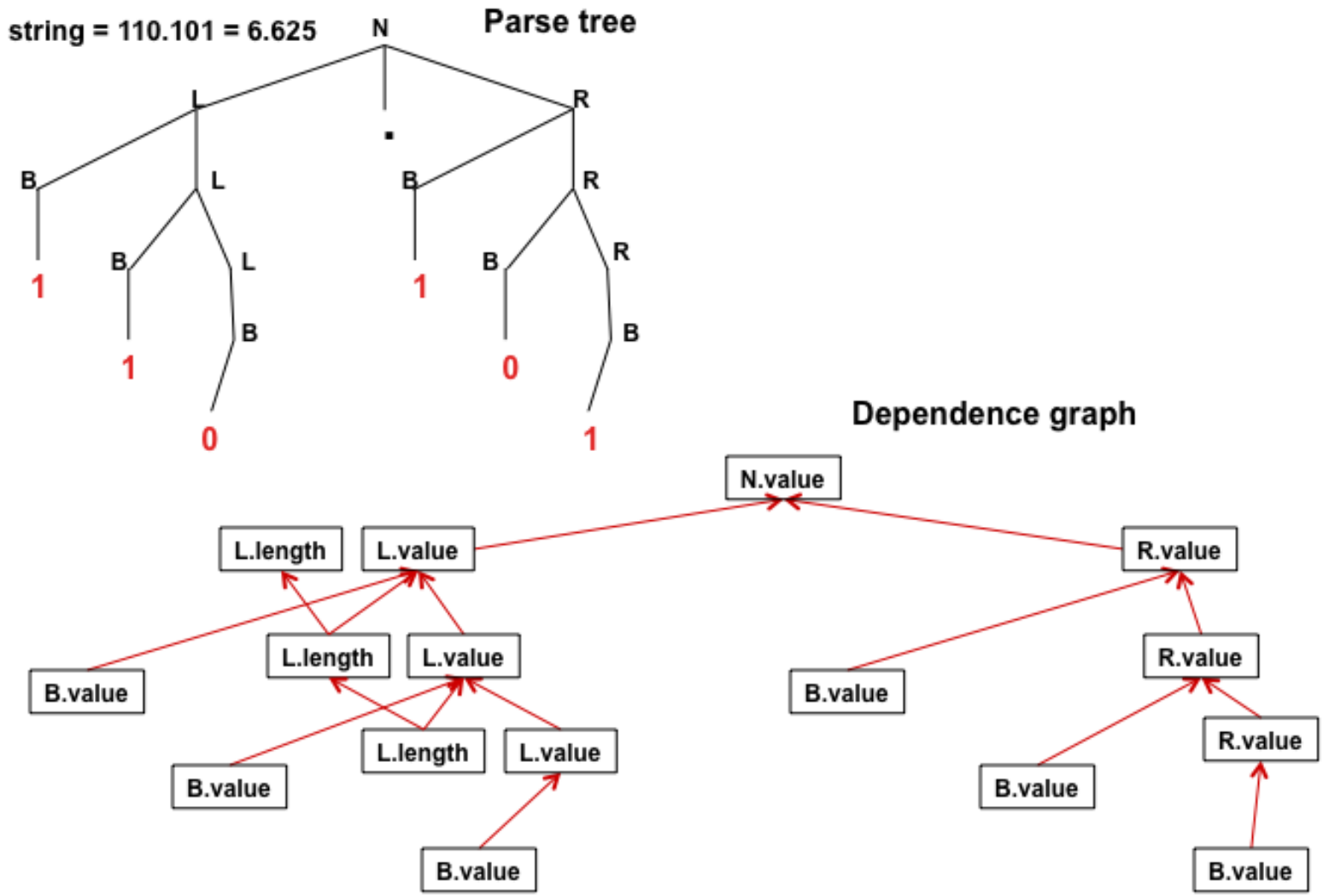
Attribute Evaluation Strategy

- Construct the parse tree
- Construct the dependence graph
- Perform topological sort on the dependence graph and obtain an evaluation order
- Evaluate attributes according to this order using the corresponding attribute evaluation rules attached to the respective productions

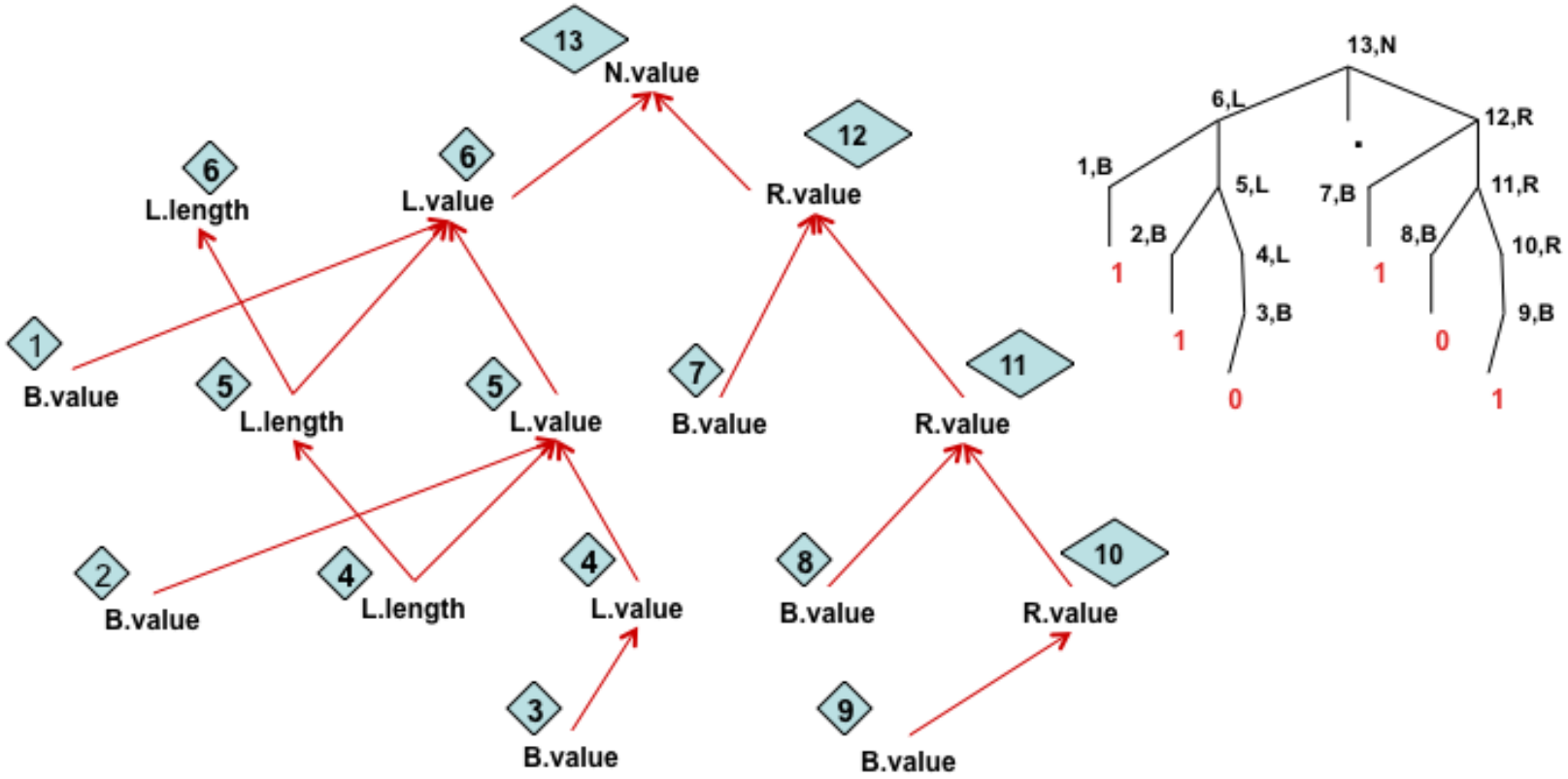
Attribute Grammar - Example 2

- AG for the evaluation of a real number from its bit-string representation
  Example: 110.101 = 6.625
- $N \rightarrow L.R,\ L \rightarrow BL \mid B,\ R \rightarrow BR \mid B,\ B \rightarrow 0 \mid 1$
- $AS(N) = AS(R) = AS(B) = \{value\uparrow: real\}$,
  $AS(L) = \{length\uparrow: integer,\ value\uparrow: real\}$
  1. $N \rightarrow L.R\ \{N.value\uparrow := L.value\uparrow + R.value\uparrow\}$
  2. $L \rightarrow B\ \{L.value\uparrow := B.value\uparrow;\ L.length\uparrow := 1\}$
  3. $L_1 \rightarrow BL_2\ \{L_1.length\uparrow := L_2.length\uparrow + 1;$
     $L_1.value\uparrow := B.value\uparrow * 2^{L_2.length\uparrow} + L_2.value\uparrow\}$
  4. $R \rightarrow B\ \{R.value\uparrow := B.value\uparrow / 2\}$
  5. $R_1 \rightarrow BR_2\ \{R_1.value\uparrow := (B.value\uparrow + R_2.value\uparrow)/2\}$
  6. $B \rightarrow 0\ \{B.value\uparrow := 0\}$
  7. $B \rightarrow 1\ \{B.value\uparrow := 1\}$

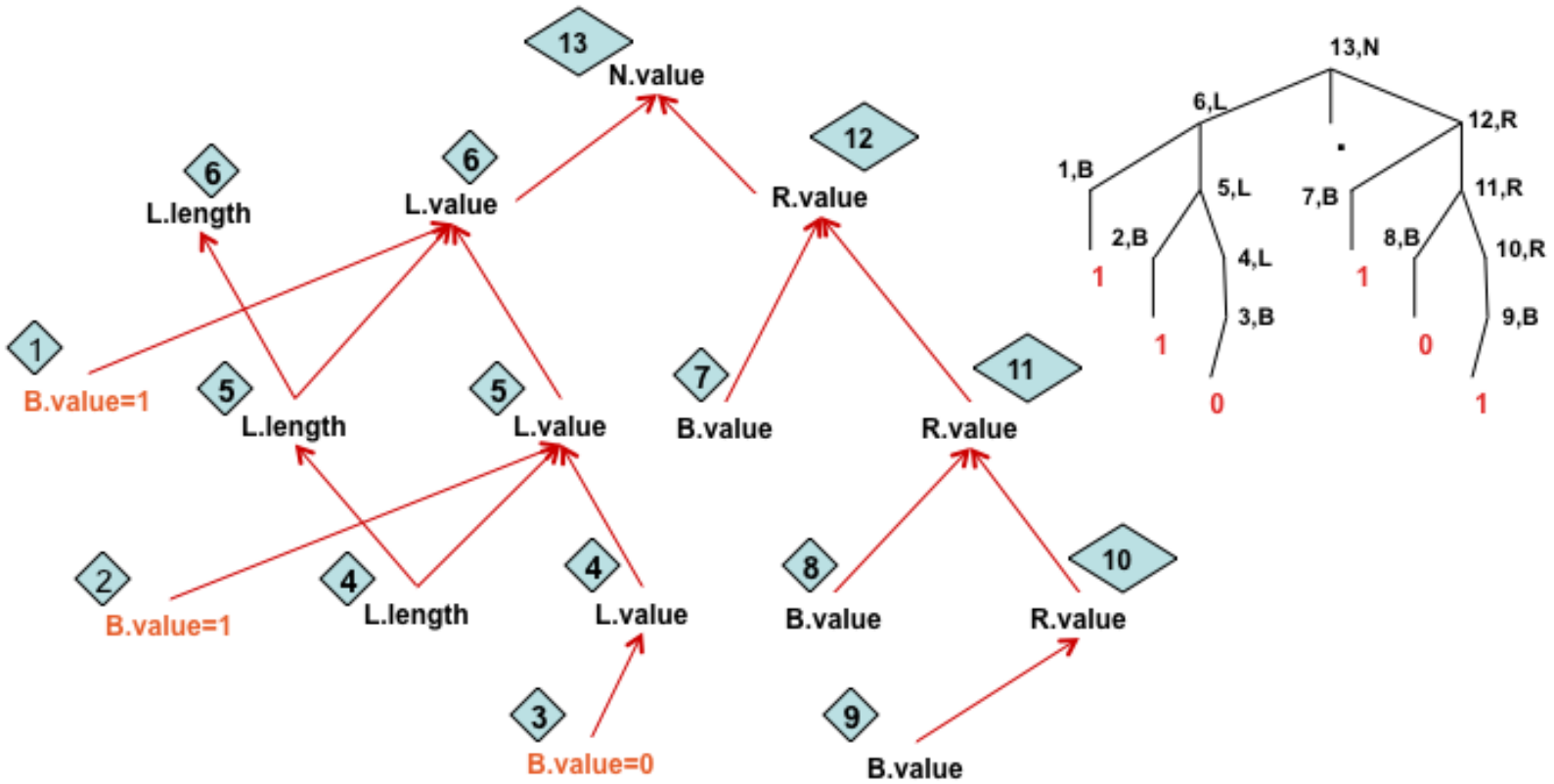
Dependence Graph for Example 2



Attribute Evaluation for Example 2 - 1



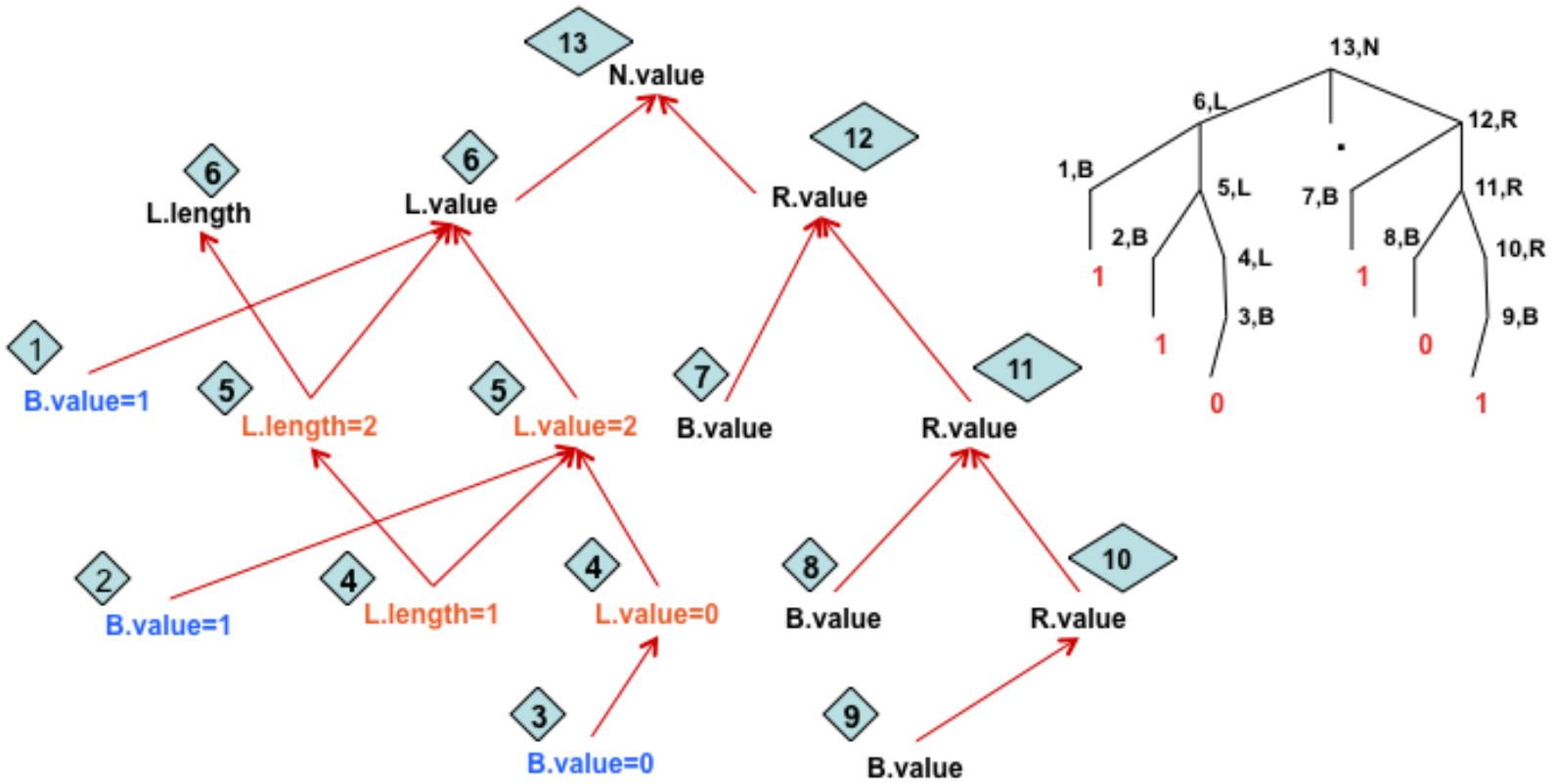
Attribute Evaluation for Example 2 - 2



Nodes 1,2: $B \rightarrow 1$ $\{B.value \uparrow := 1\}$

Node 3: $B \rightarrow 0$ $\{B.value \uparrow := 0\}$

Attribute Evaluation for Example 2 - 3

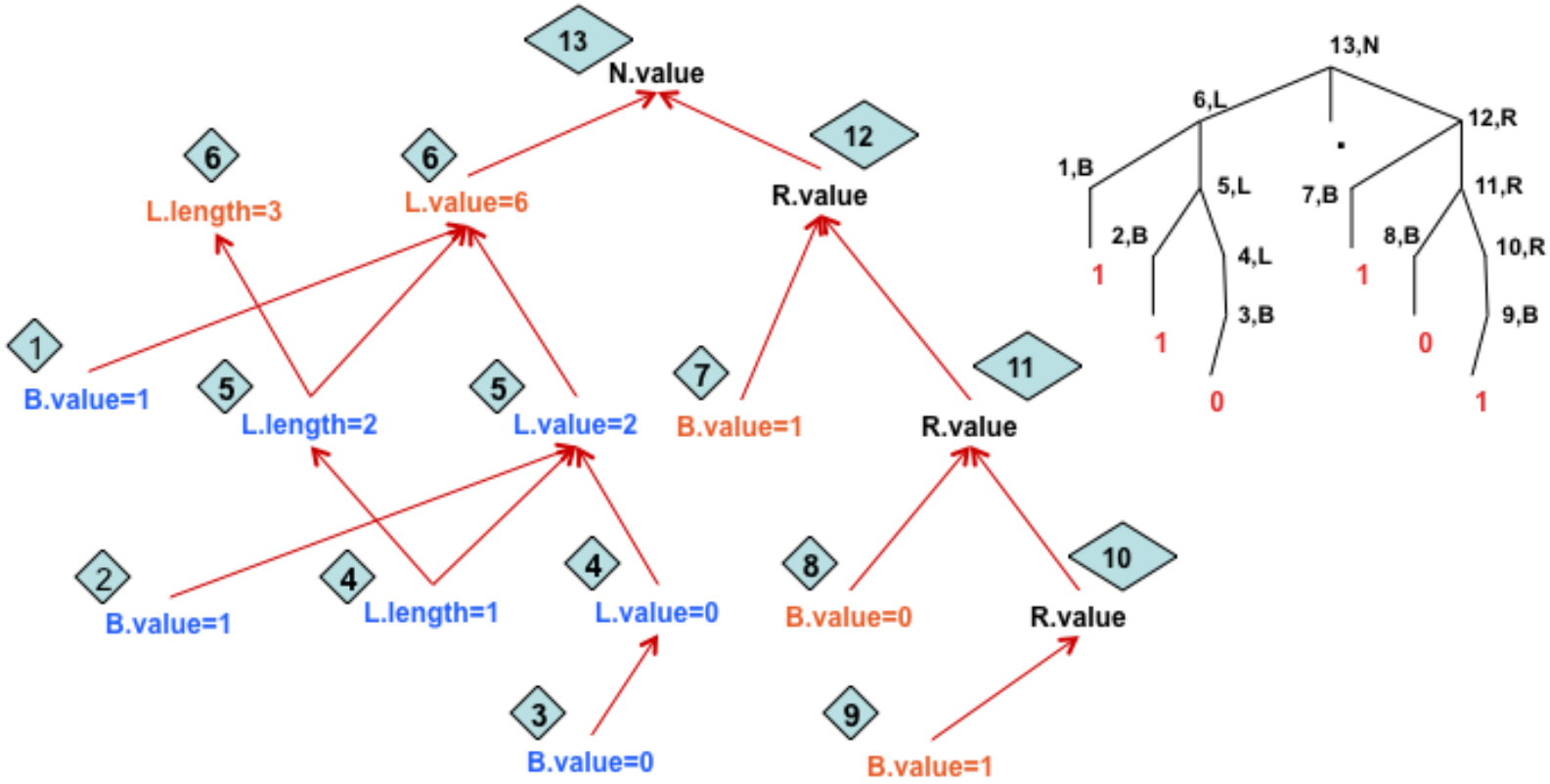


Node 4: $L \rightarrow B$ $\{L.value\uparrow := B.value\uparrow;\ L.length\uparrow := 1\}$

Node 5: $L_1 \rightarrow BL_2$ $\{L_1.length\uparrow := L_2.length\uparrow + 1;$

$L_1.value\uparrow := B.value\uparrow * 2^{L_2.length\uparrow} + L_2.value\uparrow\}$

Attribute Evaluation for Example 2 - 4

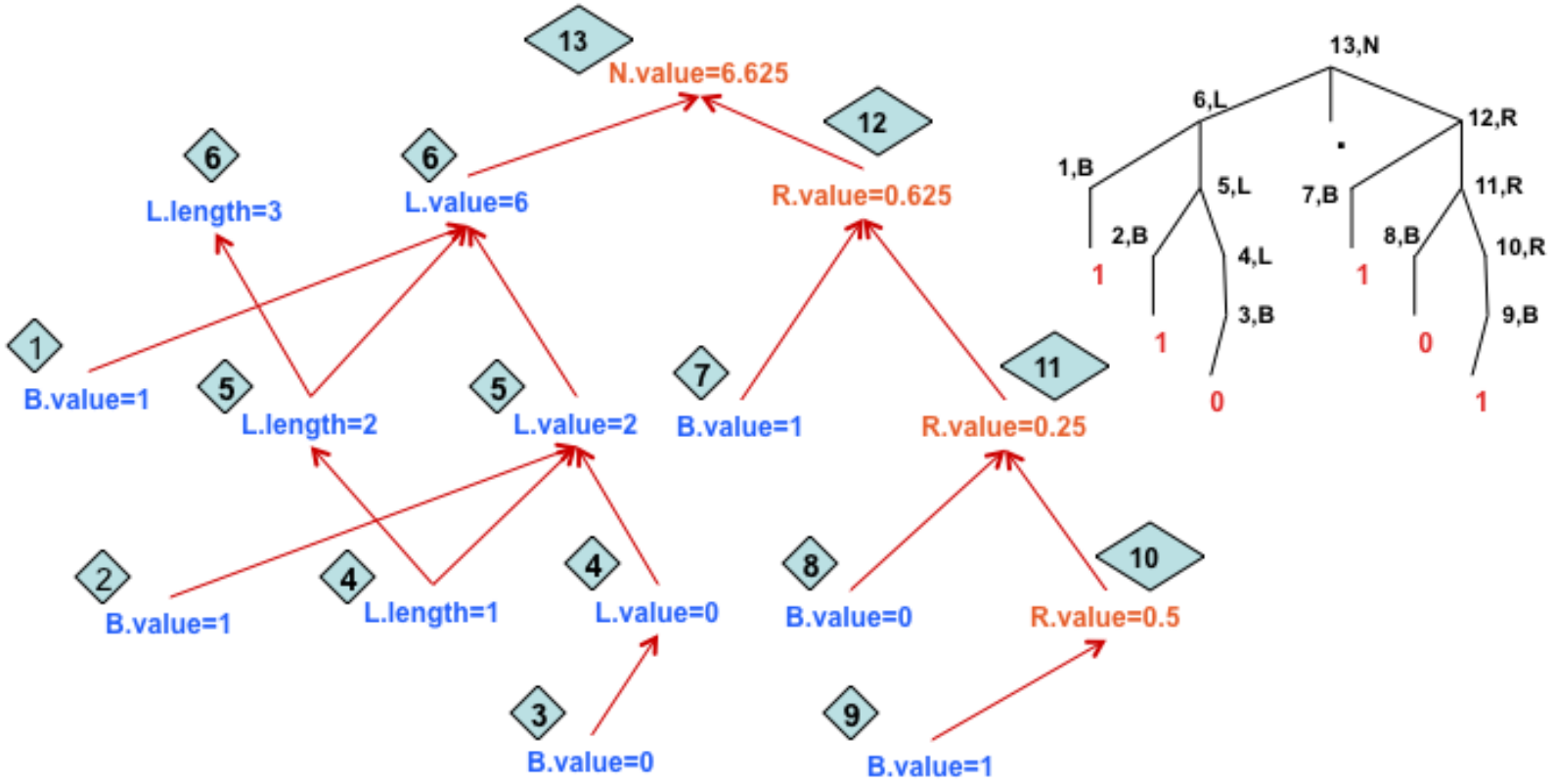


Node 6: $L_1 \rightarrow BL_2$ $\{L_1.length\uparrow := L_2.length\uparrow + 1;$
$L_1.value\uparrow := B.value\uparrow * 2^{L_2.length\uparrow} + L_2.value\uparrow\}$

Nodes 7,9: $B \rightarrow 1$ $\{B.value\uparrow := 1\}$

Node 8: $B \rightarrow 0$ $\{B.value\uparrow := 0\}$

Attribute Evaluation for Example 2 - 5



Node 10: $R \rightarrow B \; \{R.value\uparrow := B.value\uparrow /2\}$

Nodes 11,12:

$R_1 \rightarrow BR_2 \; \{R_1.value\uparrow := (B.value\uparrow + R_2.value\uparrow)/2\}$

Node 13: $N \rightarrow L.R \; \{N.value\uparrow := L.value\uparrow + R.value\uparrow\}$