3 Syntax Directed Translation

1. Consider the following Syntax Directed Translation Scheme (SDTS), with non-terminals S,A and terminals a,b.

   $S \rightarrow aAb\ print1$
   $S \rightarrow a\ print2$
   $A \rightarrow Sba\ print3$

   Using the above SDTS , the output printed by a bottom-up parser, for the input *aaababbab* is:
   (A) 1 3 2 3 2
   (B) 2 2 3 3 2
   (C) 2 3 1 3 1
   (D) 2 1 3 3 1

2. A shift reduce parser carries out the actions specified within braces immediately after reducing with the corresponding rule of grammar:

   $S \rightarrow xxW\ \ print``3''$
   $S \rightarrow y\ print``2''$
   $W \rightarrow Sz\ print``1''$

   What is the translation of xxxxyzz using the syntax directed translation scheme described by the above rules?
   (A) 23131
   (B) 11233
   (C) 21313
   (D) 33211

3. Consider the syntax-directed translation schema (SDTS) shown below:

   $E \rightarrow E + Eprint`` + 3''$
   $E \rightarrow E * Eprint``.''$
   $E \rightarrow idprintid.name$
   $E \rightarrow (E)$

   An LR-parser executes the actions associated with the productions immediately after a reduction by the corresponding production. Draw the parse tree and write the translation for the sentence: $(a+b)*(c+d)$, using SDTS given above.

4. Consider the following grammar and corresponding Syntax directed defi-

nition.

$S \to L.L \quad \{S.val = L1.val + L2.val * 2^{-L2.count}\}$
$\quad |L \quad \{S.val = L.val\}$
$L \to LB \quad \{L.dv = 2 * L1.val + B.val; L.count = L1.count + B.count\}\}$
$\quad |B \quad \{L.val = B.val; L.count = B.count\}$
$B \to 0 \quad \{B.val = 0; B.count = 1\}$
$\quad |1 \quad \{B.value = 1; B.count = 1\}$

What will be the value of $S.val$ when the bit-stream 1100101.11001 is fed to it as input?
(A) 141.125
(B) 145.625
(C) 101.78125
(D) 121.625

5. Consider the syntax directed translation scheme (SDTS) given below. Assume attribute evaluation with bottom-up parsing, i.e., attributes are evaluated immediately after a reduction.

$E \to E * T \ \{E.val = E.val * T.val\}$
$E \to T \ \{E.val = T.val\}$
$T \to F - T \ \{T.val = F.val - T.val\}$
$T \to F\{T.val = F.val\}$
$F \to 2\{F.val = 2\}$
$F \to 4\{F.val = 4\}$

Using this SDTS, construct a parse tree for the expression $4 - 2 - 4 * 2$ and also compute its E.val.

6. In a bottom-up evaluation of a syntax directed definition, inherited attributes can
(A) always be evaluated
(B) be evaluated only if the definition is L-attributed
(C) be evaluated only if the definition does not have synthesized attributes
(D) none of the above

7. Consider the following grammar and its corresponding SDD. The nonterminals are B and D, the terminals are 0 and 1.

$B \to DB \quad \{B.pos := B1.pos + 1\ \}$
$\quad B.val := B1.val + D.val$
$\quad D.pow := B1.pos$
$B \to D \quad \{B.pos := 1\}$
$\quad B.val := D.val; \ \ D.pow := 0\}$

$D \rightarrow 0$ $\{D.val := 0\}$
$D \rightarrow 1$ $\{D.val := 2^{D.pow}\}$
Here $B.pos$, $B.val$ and $D.val$ are synthesized attributes and $D.pow$ is an inherited attribute. For the input bit pattern of 101100101 the final values of $B.pos$ and $B.val$ respectively are:
(A) 10, 478
(B) 8, 239
(C) 7, 269
(D) 9, 357

8. To compute the values of the attributes at each node of an annotated parse tree we need to define an evaluation order. This is done with the help of a dependency graph drawn using the original parse tree. What is the time complexity to generate such a Dependency graph?
(A) $\theta(n)$
(B) $\theta(nlogn)$
(C) $\theta(n^2)$
(D) $\theta(n^3)$

9. Given below the syntax-directed definition (SDD), construct the annotated parse tree for the input expression: "int a, b".
$D \rightarrow TL$ $L.inh = T.type$
$T \rightarrow int$ $T.type = integer$
$T \rightarrow float$ $T.type = float$
$L \rightarrow L1, id$ $L1.inh = L.inh$
$addType(id.entry, L.inh)$
$L \rightarrow id$ $addType(id.entry, L.inh)$

10. Consider the following SDD:
$S \rightarrow N$ $N.s := 0;\ S.v := N.v$
$N \rightarrow TN$ $T.s := N_0.s;\ N_1.s := N_0.s + 1;\ N_0.v := T.v + N_1.v$
$N \rightarrow T$ $T.s := N.s;\ N.v := T.v$
$T \rightarrow 0$ $T.v := 0$
$T \rightarrow 1$ $T.v := 3^{T.s}$
$T \rightarrow 2$ $T.v := 2 * 3^{T.s}$

Answer the following questions:
(i) Construct the parse tree for the string 210
(ii) For each node classify its attributes as synthesized or inherited.
(iii) Annotate each nonterminal node with its attributes values.
(iv) What is this attributed grammar doing?
(v) What is the value of the root node $N.v$ for strings 1220 and 111.

11. Consider the following SDD over the grammar defined by $G = (\{S, A, Sign\}, S, \{`,', `-', `+', `n'\}, P)$ with P the set of production and the corresponding semantic rules depicted below. There is a special terminal symbol "n" that is lexically matched by any string of one numeric digit and whose value is the numeric value of its decimal representation. For the non-terminal symbols in G we have defined two attributes, sign and value. The non-terminal A has these two attributes whereas S only has the value attribute and Sign only has the sign attribute.

$S \rightarrow ASign$ || $S.val = A.val; A.sign = Sign.sign; print(A.val);$
$Sign \rightarrow +$ || $Sign.sign = 1$
$Sign \rightarrow -$ || $Sign.sign = 0$
$A \rightarrow n$ || $A.val = value(n)$
$A \rightarrow A1, n$ || $A1.sign = A.sign;$
$if(A.sign = 1) then$
$A.val = min(A1.val, value(n));$
$else A.val = max(A1.val, value(n));$

For this SDD answer to the following questions:
(i) Explain the overall operation of this syntax-directed definition and indicate which of the attributes are either synthesized or inherited.
(ii) Give an attributed parse tree for the source string "5, 2, 3−″ and evaluate the attributes in the attributed parse tree depicting the order in which the attributes need to be evaluated (if more than one order is possible indicate one.)

12. Consider an attributed grammar shown below:

PRODUCTION	SEMANTIC RULE
$L \rightarrow E\mathsf{n}$	$print(E.val)$
$E \rightarrow E_1 + T$	$E.val := E_1.val + T.val$
$E \rightarrow T$	$E.val := T.val$
$T \rightarrow T_1 * F$	$T.val := T_1.val * F.val$
$T \rightarrow F$	$T.val := F.val$
$F \rightarrow (E)$	$F.val := E.val$
$F \rightarrow \mathsf{digit}$	$F.val := \mathsf{digit}.lexval$

Which of the following statements about this grammar is/are correct?
(i) Semantic rules in this grammar are evaluated by a bottom-up approach or by PostOrder traversal of the parse-tree.
(ii) The annotated parse-tree for the input $3 * 5 + 4n$ has height seven.
(iii) The annotated parse-tree for the input $3 * 5 + 4 + 2 * 3n$ value of has

root $E.val$ as 25.
(A) (i) only
(B) (iv) only
(C) (iii) and (iv) only
(C) (ii) and (iii) only
(D) (i) and (iii) only
(E) (ii) and (iv) only

13. Consider following syntax directed definition:

$D \rightarrow TL$	||	$L.in := T.type$
$T \rightarrow int$	||	$T.type := integer$
$T \rightarrow real$	||	$T.type := real$
$L \rightarrow L_1, id$	||	$L_1.in := L_1.in;\ addtype(id.entry, L.in)$
$L \rightarrow id$	||	$addtype(id.entry, L.in)$

Answer following questions:
(i) List all synthesied and inherited attributes.
(ii) Draw the annotated parse-tree for the input *real id*1, *id*2, *id*3.
(iii) Draw the parse tree for the string: *int id*1, *id*2, *real id*3, *id*4.

14. Consider the Translation scheme shown below:
$A \rightarrow a\ \{output(a)\}$
$A \rightarrow (A)\ \{\ \}$
$A \rightarrow AfA\ \{output(f)\}$
where 'f' can be replaced by any arithmetic operator (i.e. $+, -, *, /..$), 'A' is a nonterminal and 'a' represents any integer value.
Draw the parse tree for expression:
$a + (a + d) * (b - c)$ and evaluate the output generated.

15. Consider following statements about L-attributed grammar:
(i) Its attributes can always be evaluated in depth-first order.
(ii) Its attributes can always be evaluated in breadth-first order.
(iii) L-attributed definitions include all syntax-directed definitions based on LL(1) grammars.
(iv) No right-to-left dependencies between attribute occurrences in the productions.
Which of the above statements is/are not true about L-attributed grammar?
(A) (i) only
(B) (iv) only
(C) (iii) and (iv) only
(C) (ii) and (iii) only
(D) (i) and (iii) only

(E) (ii) and (iv) only

16. Consider following grammar and its semantic rules:

$A \to LM$ $\{L.i := f1(A.i)\}$
$M.i := f2(L.s)$
$A.s := f3(M.s)$
$A \to QR$ $\{R.i := f4(A.i)\}$
$Q.i := f5(R.s)$
$A.i := f6(Q.s)$

The syntax directed definition os the above grammar is:
(A) S-attributed but not L-attributed
(B) L-attributed but not S-attributed
(C) Both S-attributed and L-attributed
(D) None

17. Which of the following statements about translation scheme and L-attributed definition are correct?
(A) L-attributed definition offers treatment of inherited attributed which is a less powerful theoretical concept while translation scheme is closer to a true computer program, leading to a more practical concept.
(B) L-attributed definition offers treatment of inherited attributed, leading to a less powerful computer program while translation scheme is a stronger theoretical concept.
(C) L-attributed definition offers treatment of inherited attributed, leading to a more powerful theoretical concept while translation scheme is closer to a true computer program, leading to a more practical concept.
(D) None

18. Consider following statements about translation scheme:
(i) A Translation scheme is a context-free grammar in which semantic rules are embedded within the right sides of the productions.
(ii) A translation scheme is like a syntax-directed definition, except that the order of evaluation of the semantic rules is explicitly shown.
(iii) When drawing a parse tree for a translation scheme, an action is indicated by constructing an extra child, connected by a dashed line to the node of the production.
(iv) The order in which the actions are executed is the order in which they appear during a depth-first traversal of a parse tree.
Which of the above statements about translation scheme is/are true?
(A) (i) only
(B) (i) and (ii) only
(C) (ii) and (iii) only
(D) (i), (ii) and (iii) only

(E) (i), (ii), (iii) and (iv) only

19. Which of the following statements about transforming an L-attributed grammar into a translation scheme is/are correct?
(i)The inherited attributes of X_j must be computed in actions occurring to the left of X_j in the translation.
(ii) An action can refer to a synthesized attribute $X_j.s$ of X_j only if X_j appears to the left of this action.
(iii) A synthesized attribute of A should be computed in an action occurring at the end of the translation.
(A) (i) only
(B) (ii) only
(C) (iii) only
(D) (i) and (ii) only
(E) (ii) and (iii) only
(F) (i) and (iii) only
(G) (i), (ii) and (iii) only

20. YACC parser is an LALR parser. Consider following statements about YACC:
(i) YACC parser evaluates the actions by a depth-first traversal of the parse tree.
(ii) A YACC program is a Syntax directed definition.
(iii) An action occurring inside the right side of a production can be replaced by a new dummy non-terminal that generates a nonempty string and produces the corresponding action.
Which of the following statements about YACC is true?
(A) (i) only
(B) (ii) only
(C) (iii) only
(D) (i) and (ii) only
(E) (ii) and (iii) only
(F) (i) and (iii) only
(G) (i), (ii) and (iii) only

21. Consider the following translation scheme fragment:
A $\longmapsto$ B1 {e1} B2 {e2} B3 {e3}
Which of the following is equivalent to the given fragment of translation scheme?

(A)
A $\longmapsto$ B1 C1 B2 C2 B3 C3
C1 $\longmapsto \varepsilon$
C2 $\longmapsto \varepsilon$

C3 $\longmapsto \varepsilon$

(B)
A $\longmapsto$ B1 C1 B2 C2 B3 C3
C1 $\longmapsto$ {e1}
C2 $\longmapsto$ {e2}
C3 $\longmapsto$ {e3}

(C)
A $\longmapsto$ C1 C2 C3
C1 $\longmapsto \varepsilon$ B1{e1}
C2 $\longmapsto \varepsilon$ B2{e2}
C3 $\longmapsto \varepsilon$ B3{e3}

(D)
A $\longmapsto$ B1 C1 B2 C2 B3 C3
C1 $\longmapsto \varepsilon$ {e1}
C2 $\longmapsto \varepsilon$ {e2}
C3 $\longmapsto \varepsilon$ {e3}