Compiler Design

Parsers

QUESTIONS

1. Consider the context free grammar

S → SS + |SS*|a

The language generated by the grammar is:

a) L={ Postfix expression consisting of digits , plus and multiplication sign}

b) L={ Prefix expression consisting of digits , plus and multiplication sign}

c) L={ Infix expression consisting of digits , plus and multiplication sign}

d) None of these

2. Left factoring is the process of factoring out

a) Common prefixes that appear in two or more productions of the same non-terminal

b) Predictive parsing

c) Suffixes of the alternative of the grammar rule

d) None of these

3. Which parser detects error faster?

a) LR(0)
b) LALR(1)
c) SLR(1)
d) None of these

4. Which one of the following grammar generates the language which depicts right associative list of identifiers separated by commas

a) expr → expr, id
expr → id

b) expr → id, expr
expr → id

c) expr → id

d) None of these

5. For the given grammar what is the precedence & associativity of the operators

$$\begin{aligned} E &\rightarrow E*F \\ &\quad |F+E \\ &\quad |F \\ F &\rightarrow F-F \\ &\quad |id \end{aligned}$$

a) + and * is of the same precedence & *is the left associative while + is right associative

b) * has higher precedence than + and * is left associative while + is right associative

c) + has higher precedence than * and * and + are both left associative

d) * has higher precedence than + and * and + are both left associative

6. The difference between LR(0) and SLR(1) is:

a) They differ in placement of both shift and reduce moves.

b) They differ in placement of shift moves.

c) They differ in placement of reduce moves.

d) Both are same.

7. The first and follow for the grammar below is :

$S \rightarrow aSbS|bSaS|\in$

a) FIRST(S) = {a,b,∈}
   FOLLOW(S) = {b,$}
b) FIRST(S) = {a,b}
   FOLLOW(S) = {a,b,$}
c) FIRST(S) = {a,b,∈}
   FOLLOW(S) = {a,b,$}
d) FIRST(S) = {a,b,∈}
   FOLLOW(S) = {a,b}

8. Consider the following grammar G

G : S → EF
E → a|∈
F → abF|ac

Which of the following is true about the grammar G?

(1) G is a LL(1) grammar
(2) G is a regular grammar

a) 1 only **b)** 2 only **c)** 1 and 2 both **d)** None of these

9. The number of tokens in c statement is :

printf("i=%d, &i=%X", i, &i);

a) 3 **b)** 26 **c)** 10 **d)** 21

10. Suppose that we want to describe Java style class declarations like these using a grammar

class car extends vehicle
public class JavaIsCrazy implements Factory, Builder, Listener
public final class Zergling extends Unit implements Rush

Grammar for this is

1) S→ C
2) C→PF class identifier XY
3) P→ public
4) P →∈
5) F →final
6) X →extends identifier
7) X →∈
8) Y →implements I
9) Y →∈
10) I →identifier J
11) J→ , I (note comma before I)
12) J →∈

For reference the terminals in this grammar are: public, final, class, identifier, extends, implements

The LL(1)parsing table is given below:

	Public	final	Class	extends	implements	identifiers	,	$
S	1	1	E1					
C	2	2	2					
P	3	E2	4					
F		E3	6					
X				7	8			8
Y					E4			10
I						11		
J							E5	13

E1,E2,E3,E4,E5 shall be filled with

a) 2,6,3,9,12 **b)** 1,4,5,9,12

c) 1,4,5,11,12 **d)** 1,4,6,11,12

11. Consider the following grammar:

G : S→E|e

E→e

The grammar G is :

a) LL(1) **b)** SLR(1) **c)** LR(0) **d)** None of these

12. . Identify the correct sequence of parses arranged in decreasing order of their power:

a. CLR(1), SLR(1), LR(0), LALR(1)

b. CLR(1), LALR(1), SLR(1), LR(0)

c. LALR(1), SLR(1), CLR(1), LR(0)

d. LR(0), SLR(1), LALR(1), CLR(1)

13. Consider the following grammar

S→ABC……..1

S→X……2

S→∈…….3

Which production of the above grammar violates the condition of operator grammar?

a) 1 only **b)** 1 and 3 **c)** 3 only **d)** 1, 2 and 3

14. Consider the following grammar

S→S(S)| ∈

Which one of the following is true

1. Grammar is ambiguous
2. Grammar is unambiguous
3. The grammar will generate all strings having balanced parenthesis

a) 1 and 3 **b)** 2 and 3 **c)** 1 only **d)** 2 only

Common Data Question 15 and 16

Consider the operator precedence relation

	id	+	*	$
id		▪>	▪>	▪>
+	<▪	▪>	▪>	▪>
*	<▪	<▪	▪>	▪>
$	<▪	<▪	<▪	

15. After evaluating id1,id2,id3 give the order in which operator will be evaluated :

a) +,* **b)** *,+ **c)** any operator can be evaluated **d)** none of the above

16. Suppose we are evaluating the string id_1+id_2+id_3. Give the order in which it shall be evaluated.

a) id_1+(id_2+id_3)
b) Cannot be said
c) (id_1+id_2)+id_3
d) None of these

Common Data Question 17 and 18

17. Below is a CFG for strings of balanced parenthesis

1) S→P
2) P→(P)P
3) P→∈

The SLR(1)parsing table is :

	Action			Goto	
	(	)	$	P	S
0	S_3	E1	r_3	2	1
1			E2		
2		r_2	r_2		
3	S_3	E3	r_3	4	
4		S_5			
5	E4	r_3	r_3	6	
6		r_2	r_2		

The follow sets of non terminals will be :

a) FOLLOW(S) = { $ }
FOLLOW(P) = {),$ }
b) FOLLOW(S) = { (, $ }
FOLLOW(P) = {(,), $ }
c) FOLLOW(S) = {), (**}**
FOLLOW(P) = { (,), $}
d) FOLLOW(S) = { $ }
FOLLOW(P) = { $ }

18. The entries E1,E2,E3,E4 will be :

a) r_3, accept, S_5, r_3

b) S_3, r_4, r_5, S_3

c) r_3, accept, r_3, S_3

d) r_3, accept, r_4, r_3

19. Given the grammar

E→ E + T/T
T→ T * F/F
F→ G ↑ F/G

G→ id

Specify which operator has the highest precedence and whether it is left or right recursive?

a) + highest precedence and left recursive.
b) ↑ highest precedence and right recursive
c) ↑ lowest precedence and left recursive
d) * highest precedence and is right recursive

20. Determine whether the following grammar is :

$$S \rightarrow AS|b$$
$$A \rightarrow SA|a$$

a) LL(1) **b)** LL(0) **c)** SLR(1) **d)** None of these