

COMPILER DESIGN CSE-3151– INSEM2 SCHEME

Q1. The total number of Kernel items generated for the grammar given is _____

$E' \rightarrow E$
 $E \rightarrow E+T \mid T$

(0.5)

1. 4
2. **6
3. 7
4. 5

Q2. Let 'u' be the first symbol of the input string 'w' and X be the stack top symbol. Which of the following is TRUE w.r.t Predictive parsing algorithm when 'X' is equal to 'u'? (0.5)

1. Declare error.
2. Output the production, pop the stack and push the production to the stack.
3. **Pop the stack and let 'u' point to next symbol of 'w'.
4. None of the mentioned.

Q3. In LR-parsing program, which of the following is TRUE when (ACTION[s,a]= shift t)? (0.5)

1. Pop symbols off the stack.
2. ** Push 't' onto the stack and let 'a' point to the next input symbol.
3. Push GOTO [t,A] onto the stack.
4. Output the productions and let state 't' be the current stack top.

Q4. Consider the following augmented grammar with {#, @, <, >, a, b, c} as the set of terminals. Let I_0 be closure ($S' \rightarrow S$). The number of items in the set $GOTO(GOTO(I_0, <), <)$ is _____.

$S' \rightarrow S$
 $S \rightarrow S\#cS \mid SS \mid S@ \mid <S> \mid a \mid b \mid c$

(0.5)

1. 1
2. 6
3. 7
4. **8

Q5. To construct the Canonical LR(0) collection for a grammar, we need to define an augmented grammar for G and two functions namely:

(0.5)

1. ** CLOSURE, GOTO
2. SHIFT, REDUCE.

3. REDUCE, GOTO.
4. None of the mentioned.

Q6. For given grammar,

$P \rightarrow MN$
 $M \rightarrow Le \mid \epsilon$
 $N \rightarrow NeML \mid g$
 $L \rightarrow f \mid \epsilon$

Consider $\{e, f, g, \$\}$ as $\{0, 1, 2, 3\}$. What is the follow of M using bottom up approach?

(0.5)

1. $\{3\}$
2. $\{0, 3\}$
3. ****** $\{0, 1, 2, 3\}$
4. $\{1, 3\}$

Q7. Consider a grammar –

$E \rightarrow E+n \mid E*n \mid n$

For a sentence “ $n+n*n$ ”, the handles in the right-sentential form of the reduction are

(0.5)

1. $n, n+n$ and $n+n*n$
2. ****** $n, E+n$ and $E*n$
3. $n, E+n$ and $E+n*n$
4. $n, E+n$ and $E+E*n$

Q8. Which among the following test does NOT match the given regular expression pattern?

$\text{rege}(x(es)?|xps?)$ **(0.5)**

1. regexps
2. regexes
3. regexp
4. ******regep

Q9. Choose the correct option.

I) The states of LR(0) automaton are the sets of items from the canonical LR(0) collection.

II) If $A \rightarrow \alpha \bullet B \beta$ is in $\text{CLOSURE}(I)$ and $B \rightarrow \gamma$ is a production then add item $B \rightarrow \gamma \bullet$ **(0.5)**

1. ******I – TRUE, II – FALSE

2. I- TRUE, II – TRUE
3. I-FALSE, II – FALSE
4. I-FALSE, II- TRUE

Q10. In LR automata, leads to ONLY state change. (0.5)

1. ****Goto**
2. Shift
3. Reduce
4. Accept

Q11. For the given grammar, compute the FIRST and FOLLOW set for each non-terminal by considering 'E' as the start symbol. (2)

$E \rightarrow TE'$
 $E' \rightarrow +E \mid \epsilon$
 $T \rightarrow FT'$
 $T' \rightarrow T \mid \epsilon$
 $F \rightarrow PF'$
 $F' \rightarrow *F' \mid \epsilon$
 $P \rightarrow (E) \mid a|b|c \mid \epsilon$

	First -1M	FOLLOW-1M
E	(, a,b,c,*, +, ϵ	\$,)
E'	+, ϵ	\$,)
T	(, a,b,c,*, ϵ	\$,), +
T'	(, a,b,c,*, ϵ	\$,), +
F	(, a,b,c,*, ϵ	(, a,b,c,*, +, \$,)
P	(, a,b,c, ϵ	(, a,b,c,*, +, \$,)
F'	*, ϵ	(, a,b,c,*, +, \$,)

Q12. Illustrate with the help of suitable syntactic constructs that give rise to grammars which results in conflicts when used for Shift Reduce Parsing (2)

Two types of conflicts exist:

- 1) Shift reduce conflict: The below example shows a configuration with shift reduce conflict during parsing. This can be solved by giving priority to shift action.

$$\begin{array}{lcl} \text{stmt} & \rightarrow & \text{if expr then stmt} \\ & | & \text{if expr then stmt else stmt} \\ & | & \text{other} \end{array}$$

If we have a shift-reduce parser in configuration

STACK	INPUT
... if expr then stmt	else ... \$

- 2) Reduce-Reduce conflicts: In the below grammar we see that both stmt and expr begin with id and so the following configuration gives rise to reduce-reduce conflict. The solution is to change one of them to procid.

STACK	INPUT
... id (id	, id) ...

- (1) stmt \rightarrow id (parameter_list)
- (2) stmt \rightarrow expr := expr
- (3) parameter_list \rightarrow parameter_list , parameter
- (4) parameter_list \rightarrow parameter
- (5) parameter \rightarrow id
- (6) expr \rightarrow id (expr_list)
- (7) expr \rightarrow id
- (8) expr_list \rightarrow expr_list , expr
- (9) expr_list \rightarrow expr

STACK	INPUT
... procid (id	, id) ...

Examples- 1M, Explanation-1M

Q13. Consider the following CFG,

$$\begin{array}{lcl} P & \rightarrow & S \\ S & \rightarrow & \text{stmt} \mid \{B \\ B & \rightarrow & \} \mid S\} \mid S;B \end{array}$$

- (a) Construct a LR(0) DFA and construct SLR(1) parse table.
- (b) Assuming that an SLR(1) parser resolves shift-reduce conflicts by choosing to shift, show the operation of such a parser on the input string {stmt;} (3)



	2	3	4	5	6	7	8	9	10
1	24			25		1	5		
2		41	41		41				
3		42	42		42				
4	26	26		53		6	10		
5		44	44		54				
6		47	47						
7		51	51		51				
8	46	46		52		7	9		
9		55	55		56				
10		57	57		57				

$sum(1) = 5$
 $sum(2) = 2 + 1 + 9$
 $sum(3) = 1 + 1 + 5$

State	Symbol	State	Action
0		1	Shift
1	2	2	Shift
2	3	3	Shift
3	4	4	Shift
4	5	5	Shift
5	6	6	Shift
6	7	7	Shift
7	8	8	Shift
8	9	9	Shift
9	10	10	Shift
10	11	11	Shift
11	12	12	Shift
12	13	13	Shift
13	14	14	Shift
14	15	15	Shift
15	16	16	Shift
16	17	17	Shift
17	18	18	Shift
18	19	19	Shift
19	20	20	Shift
20	21	21	Shift
21	22	22	Shift
22	23	23	Shift
23	24	24	Shift
24	25	25	Shift
25	26	26	Shift
26	27	27	Shift
27	28	28	Shift
28	29	29	Shift
29	30	30	Shift
30	31	31	Shift
31	32	32	Shift
32	33	33	Shift
33	34	34	Shift
34	35	35	Shift
35	36	36	Shift
36	37	37	Shift
37	38	38	Shift
38	39	39	Shift
39	40	40	Shift
40	41	41	Shift
41	42	42	Shift
42	43	43	Shift
43	44	44	Shift
44	45	45	Shift
45	46	46	Shift
46	47	47	Shift
47	48	48	Shift
48	49	49	Shift
49	50	50	Shift
50	51	51	Shift
51	52	52	Shift
52	53	53	Shift
53	54	54	Shift
54	55	55	Shift
55	56	56	Shift
56	57	57	Shift
57	58	58	Shift
58	59	59	Shift
59	60	60	Shift
60	61	61	Shift
61	62	62	Shift
62	63	63	Shift
63	64	64	Shift
64	65	65	Shift
65	66	66	Shift
66	67	67	Shift
67	68	68	Shift
68	69	69	Shift
69	70	70	Shift
70	71	71	Shift
71	72	72	Shift
72	73	73	Shift
73	74	74	Shift
74	75	75	Shift
75	76	76	Shift
76	77	77	Shift
77	78	78	Shift
78	79	79	Shift
79	80	80	Shift
80	81	81	Shift
81	82	82	Shift
82	83	83	Shift
83	84	84	Shift
84	85	85	Shift
85	86	86	Shift
86	87	87	Shift
87	88	88	Shift
88	89	89	Shift
89	90	90	Shift
90	91	91	Shift
91	92	92	Shift
92	93	93	Shift
93	94	94	Shift
94	95	95	Shift
95	96	96	Shift
96	97	97	Shift
97	98	98	Shift
98	99	99	Shift
99	100	100	Shift
100	101	101	Shift
101	102	102	Shift
102	103	103	Shift
103	104	104	Shift
104	105	105	Shift
105	106	106	Shift
106	107	107	Shift
107	108	108	Shift
108	109	109	Shift
109	110	110	Shift
110	111	111	Shift
111	112	112	Shift
112	113	113	Shift
113	114	114	Shift
114	115	115	Shift
115	116	116	Shift
116	117	117	Shift
117	118	118	Shift
118	119	119	Shift
119	120	120	Shift
120	121	121	Shift
121	122	122	Shift
122	123	123	Shift
123	124	124	Shift
124	125	125	Shift
125	126	126	Shift
126	127	127	Shift
127	128	128	Shift
128	129	129	Shift
129	130	130	Shift
130	131	131	Shift
131	132	132	Shift
132	133	133	Shift
133	134	134	Shift
134	135	135	Shift
135	136	136	Shift
136	137	137	Shift
137	138	138	Shift
138	139	139	Shift
139	140	140	Shift
140	141	141	Shift
141	142	142	Shift
142	143	143	Shift
143	144	144	Shift
144	145	145	Shift
145	146	146	Shift
146	147	147	Shift
147	148	148	Shift
148	149	149	Shift
149	150	150	Shift
150	151	151	Shift
151	152	152	Shift
152	153	153	Shift
153	154	154	Shift
154	155	155	Shift
155	156	156	Shift
156	157	157	Shift
157	158	158	Shift
158	159	159	Shift
159	160	160	Shift
160	161	161	Shift
161	162	162	Shift
162	163	163	Shift
163	164	164	Shift
164	165	165	Shift
165	166	166	Shift
166	167	167	Shift
167	168	168	Shift
168	169	169	Shift
169	170	170	Shift
170	171	171	Shift
171	172	172	Shift
172	173	173	Shift
173	174	174	Shift
174	175	175	Shift
175	176	176	Shift
176	177	177	Shift
177	178	178	Shift
178	179	179	Shift
179	180	180	Shift
180	181	181	Shift
181	182	182	Shift
182	183	183	Shift
183	184	184	Shift
184	185	185	Shift
185	186	186	Shift
186	187	187	Shift
187	188	188	Shift
188	189	189	Shift
189	190	190	Shift
190	191	191	Shift
191	192	192	Shift
192	193	193	Shift
193	194	194	Shift
194	195	195	Shift
195	196	196	Shift
196	197	197	Shift
197	198	198	Shift
198	199	199	Shift
199	200	200	Shift
200	201	201	Shift
201	202	202	Shift
202	203	203	Shift
203	204	204	Shift
204	205	205	Shift
205	206	206	Shift
206	207	207	Shift
207	208	208	Shift
208	209	209	Shift
209	210	210	Shift
210	211	211	Shift
211	212	212	Shift
212	213	213	Shift
213	214	214	Shift
214	215	215	Shift
215	216	216	Shift
216	217	217	Shift
217	218	218	Shift
218	219	219	Shift
219	220	220	Shift
220	221	221	Shift
221	222	222	Shift
222	223	223	Shift
223	224	224	Shift
224	225	225	Shift
225	226	226	Shift
226	227	227	Shift
227	228	228	Shift
228	229	229	Shift
229	230	230	Shift
230	231	231	Shift
231	232	232	Shift
232	233	233	Shift
233	234	234	Shift
234	235	235	Shift
235	236	236	Shift
236	237	237	Shift
237	238	238	Shift
238	239	239	Shift
239	240	240	Shift
240	241	241	Shift
241	242	242	Shift
242	243	243	Shift
243	244	244	Shift
244	245	245	Shift
245	246	246	Shift
246	247	247	Shift
247	248	248	Shift
248	249	249	Shift
249	250	250	Shift
250	251	251	Shift
251	252	252	Shift
252	253	253	Shift
253	254	254	Shift
254	255	255	Shift
255	256	256	Shift
256	257	257	Shift
257	258	258	Shift
258	259	259	Shift
259	260	260	Shift
260	261	261	Shift
261	262	262	Shift
262	263	263	Shift
263	264	264	Shift
264	265	265	Shift
265	266	266	Shift
266	267	267	Shift
267	268	268	Shift
268	269	269	Shift
269	270	270	Shift
270	271	271	Shift
271	272	272	Shift
272	273	273	Shift
273	274	274	Shift
274	275	275	Shift
275	276	276	Shift
276	277	277	Shift
277	278	278	Shift
278	279	279	Shift
279	280	280	Shift
280	281	281	Shift
281	282	282	Shift
282	283	283	Shift
283	284	284	Shift
284	285	285	Shift
285	286	286	Shift
286	287	287	Shift
287	288	288	Shift
288	289	289	Shift
289	290	290	Shift
290	291	291	Shift
291	292	292	Shift
292	293	293	Shift
293	294	294	Shift
294	295	295	Shift
295	296	296	Shift
296	297	297	Shift
297	298	298	Shift
298	299	299	Shift
299	300	300	Shift
300	301	301	Shift
301	302	302	Shift
302	303	303	Shift
303	304	304	Shift
304	305	305	Shift
305	306	306	Shift
306	307	307	Shift
307	308	308	Shift
308	309	309	Shift</

