CSCE 4650/5650 ASSIGNMENT #3 SOLUTION

Consider the following BNF grammar G:

expression ::= identifier | `(` expression `)` | `apply` identifier `to` expression

1. Construct the set of LR(1) items for G (the original grammar given above).

   We first rewrite the grammar as follows:

   E → E
   E → id | (E) | apply id to E

   The item sets are as follows:

```
I0   :  E → • E, $
        E → • id, $
        E → • (E), $
        E → • apply id to E, $
I1   :  E → E •, $                 GOTO (I0, E)
I2   :  E → id •, $                GOTO (I0, id)
I3   :  E → ( • E), $              GOTO (I0, (
        E → • id, )
        E → • (E), )
        E → • apply id to E, )
I4   :  E → apply • id to E, $     GOTO (I0, apply)
I5   :  E → (E • ), $              GOTO (I3, E)
I6   :  E → id •, )                GOTO (I3, id), GOTO (I7, id)
I7   :  E → ( • E), )              GOTO (I3, (), GOTO (I7, (
        E → • id, )
        E → • (E), )
        E → • apply id to E, )
I8   :  E → apply • id to E, )     GOTO (I3, apply), GOTO (I7, apply)
I9   :  E → apply id • to E, $     GOTO (I4, id)
I10  :  E → (E) •, $               GOTO (I5, ))
I11  :  E → (E • ), )              GOTO (I7, E)
I12  :  E → apply id • to E, )     GOTO (I8, id)
I13  :  E → apply id to • E, $     GOTO (I9, to)
        E → • id, $
        E → • (E), $
        E → • apply id to E, $
I14  :  E → (E) •, )               GOTO (I11, ))
I15  :  E → apply id to • E, )     GOTO (I12, to)
        E → • id, )
        E → • (E), )
        E → • apply id to E, )
I16  :  E → apply id to E •, $     GOTO (I13, E)
I17  :  E → apply id to E •, )     GOTO (I15, E)
```

2. Merge the sets of items having common cores to give the set of LALR(1) items.

   `The LALR(1) item sets and their associated states are: I0 (0), I1 (1), {I2, I6} (2), {I3, I7} (3), {I4, I8} (4), {I5, I11} (5), {I9, I12} (6), {I10, I14} (7), {I13, I15} (8), {I16, I17} (9).`

3. Construct the LALR(1) parsing table from the set of LALR(1) items.

	id	(	)	apply	to	$	E
0	s2	s3		s4			1
1						acc	
2			r1			r1	
3	s2	s3		s4			5
4	s6						
5			s7				
6					s8		
7			r2			r2	
8	s2	s3		s4			9
9			r3			r3	

4. Simplify the LALR parsing table using the compaction techniques discussed at the end of Section 4.7.

State	Input	Action	Symbol	State	Goto
0,3,8	id	s2	E	0	1
	(	s3		3	5
	apply	s4		8	9
	any	error			
1	$	accept			
	any	error			
2	any	r1			
4	id	s6			
	any	error			
5	)	s7			
	any	error			
6	to	s8			
	any	error			
7	any	r2			
9	any	r3			

5. Augment the compacted LALR table with appropriate error recovery calls and routines. Give a general explanation of how your error recovery routines work.

   `We number errors by the states in which they occur.`

```
0,3,8 - print ("id expected"); s2
1 - print ("unexpected symbol" + input); delete symbol and retry
4 - print ("id expected"); s6
```

```
5 - if input = $ print ("missing )"; s7
      else print ("unexpected symbol" + input); delete symbol and retry
6 - print ("to expected"); s8
```