1. (30 points) Regular Expressions and Context Free Grammars

i. Given the following grammar: 
   \[ S \rightarrow A B | \epsilon \quad A \rightarrow a A | \epsilon \quad B \rightarrow b B | \epsilon \]
   Identify the string (if any) that has multiple leftmost derivations:
   a) a a a
   b) a b b b
   c) \epsilon
   d) None of the above

ii. Identify the regular grammar (if any) among the following:
   a) \[ S \rightarrow (S) S | \epsilon \]
   b) \[ S \rightarrow a S | a | \epsilon \]
   c) \[ S \rightarrow A B | \epsilon \quad A \rightarrow a A | a \quad B \rightarrow b B | b \]
   d) None of the above

iii. Identify the grammar (if any) that is LL(1) among the following:
   a) \[ E \rightarrow E + T | T \quad T \rightarrow \text{id} | (E) \]
   b) \[ A \rightarrow a B | a C \quad B \rightarrow b B | b \quad C \rightarrow c C | c \]
   c) \[ S \rightarrow (S) S | \epsilon \]
   d) None of the above

iv. Consider the regular expression \((x^+ y)^? \cdot x\) where \(\Sigma = \{x,y\}\). Which of the following strings can be generated by the regular expression.
   a) y x
   b) x x y x x
   c) x x
   d) All of the above

v. Consider the regular expression \([5-7] \mid [23][0-8]\). Which of the following strings cannot be generated by the regular expression?
   a) 6
   b) 28
   c) 39
   d) None of the above

vi. Which of the following regular expressions is equivalent to given regular expression: \((a \mid b)^* c\) where \(\Sigma = \{a,b,c\}\).
   a) a b c
   b) a c | b c | c
   c) a c | b c
2. (25 points) Top Down Parsing. Given the following grammar:

\[
\begin{align*}
#1 & \quad \text{LEXP} \rightarrow \text{ATOM} \\
#2 & \quad \text{LEXP} \rightarrow \text{LIST} \\
#3 & \quad \text{ATOM} \rightarrow \text{num} \\
#4 & \quad \text{ATOM} \rightarrow \text{id} \\
#5 & \quad \text{LIST} \rightarrow ( \text{LEXP} \ LSEQ ) \\
#6 & \quad \text{LSEQ} \rightarrow \text{LEXP} \ LSEQ \\
#7 & \quad \text{LSEQ} \rightarrow \epsilon
\end{align*}
\]

i. \quad \text{TABLE [ LEXP, num ]} = \#1

ii. \quad \text{TABLE [ LEXP, ( ]} = \#2

iii. \quad \text{TABLE [ ATOM, num ]} = \#3

iv. \quad \text{TABLE [ ATOM, ( ]} = \text{error}

v. \quad \text{TABLE [ LIST, ( ]} = \#5

vi. \quad \text{TABLE [ LSEQ, num ]} = \#6

vii. \quad \text{TABLE [ LSEQ, id ]} = \#6

viii. \quad \text{TABLE [ LSEQ, ( ]} = \#6

ix. \quad \text{TABLE [ LSEQ, ) ]} = \#7

x. \quad \text{TABLE [ LSEQ, $ ]} = \text{error}
3. (45 points) **Bottom Up Parsing**: Given an incomplete SLR(1) state machine.

i. Provide *items missing* from state $S_0$: $R \to .L \ & \ L \to .a$

ii. Provide *items missing* from state $S_1$: $R \to L$

iii. Provide *items missing* from state $S_3$: $L \to a$

iv. Provide action: $\text{ACTION} [ S_0, a ] = \text{shift, S}_3$

v. Provide action: $\text{ACTION} [ S_4, a ] = \text{shift, S}_3$

vi. Provide action: $\text{ACTION} [ S_1, = ] = \text{shift, S}_4$

vii. Provide action: $\text{ACTION} [ S_4, $ ] = \text{error}$

viii. Provide action: $\text{ACTION} [ S_2, $ ] = \text{reduce, S}$\text{R / accept}$

ix. Provide action: $\text{ACTION} [ S_6, $ ] = \text{reduce, R}$\text{L}

x. Provide action: $\text{ACTION} [ S_6, = ] = \text{error}$

xi. Provide action: $\text{ACTION} [ S_5, $ ] = \text{reduce, S}$\text{L=R / accept}$

xii. Identify a state, if any, that contain a *shift-reduce* conflict?

    No (shift on = and reduce on $)$

xiii. Identify a state, if any, that contains a *reduce-reduce* conflict?

    Yes, in $S_3$ -- because $\$ \text{ is in Follow}(R)$ and Follow(L).

xiv. Will conflicts found, if any, be present in the LR(1) parser? Yes