1. (10 points) Using the method in class or in the text, construct an NFA equivalent to the following regular expression:

\([-+]?[012]+\]

**ANSWER:** There are many acceptable answers. Here is one (blank entries denote the empty set \(\emptyset\)):

<table>
<thead>
<tr>
<th>(\rightarrow q_0)</th>
<th>+</th>
<th>−</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>(\varepsilon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(q_1)</td>
<td>({q_2})</td>
<td></td>
<td></td>
<td></td>
<td>({q_1, q_3, q_5})</td>
<td></td>
</tr>
<tr>
<td>(q_2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_3})</td>
<td></td>
</tr>
<tr>
<td>(q_3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_4})</td>
<td></td>
</tr>
<tr>
<td>(q_4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_5})</td>
<td></td>
</tr>
<tr>
<td>(q_5)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_6, q_8, q_{10}})</td>
<td></td>
</tr>
<tr>
<td>(q_6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_7})</td>
<td></td>
</tr>
<tr>
<td>(q_7)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_9})</td>
<td></td>
</tr>
<tr>
<td>(q_8)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_{11}})</td>
<td></td>
</tr>
<tr>
<td>(q_{10})</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_{12}})</td>
<td></td>
</tr>
<tr>
<td>(q_{11})</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_{12}})</td>
<td></td>
</tr>
<tr>
<td>(*q_{12})</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>({q_5})</td>
<td></td>
</tr>
</tbody>
</table>

2. (40 points total) Consider the following grammar with start symbol \(S'\):

\[
S' \rightarrow S \\
S \rightarrow bSa \\
S \rightarrow Bb \\
S \rightarrow aa \\
B \rightarrow a
\]

(a) (10 points) Using the method described in class or the text, construct the set \(\{I_0, I_1, I_2, \ldots\}\) of states for an SLR (SLR(1)) parser for this grammar, defining the transition function at the same time. Stop after five states, i.e., \(\{I_0, I_1, I_2, I_3, I_4\}\). Note that in class I denoted the transition function as \(\text{trans}\). The textbook denotes the same function as \(\text{goto}\).

(b) (5 points) What is \(\text{action}[I_0, a]\)? What is \(\text{action}[I_0, b]\)?
(c) (10 points) Describe any conflicts, giving the conflicting actions.

(d) (15 points) Using the method described in class or the text, construct the full set of states along with the transition table for a canonical LR(1) parser for this grammar.

**ANSWER:**

(a)

\[
\begin{align*}
I_0 &= \text{start state :} \\
I_2 &= \text{trans}(I_0, b) : \\
I_3 &= \text{trans}(I_0, B) : \\
S' &\rightarrow .S \\
S &\rightarrow b.Sa \\
S &\rightarrow .bSa \\
S &\rightarrow .Bb \\
S &\rightarrow .aa \\
B &\rightarrow .a \\
I_4 &= \text{trans}(I_0, a) : \\
S' &\rightarrow .S \\
S &\rightarrow .Bb \\
S &\rightarrow .aa \\
B &\rightarrow .a \\
I_1 &= \text{trans}(I_0, S) : \\
S' &\rightarrow S.
\end{align*}
\]

(b) \( \text{action}[I_0, a] = \text{“shift } I_4 \text{” and action}[I_0, b] = \text{“shift } I_2. \)"

(c) There are no conflicts. It may appear that \( \text{action}[I_4, a] \) creates a conflict (either “shift \( \text{trans}(I_4, a) \)” or “reduce \( B \rightarrow a \)”), but the second action does not apply, because \( a \notin \text{FOLLOW}(B) \).

[NOTE: I meant for the fourth production to be \( S \rightarrow ab \), which \textit{would} produce a conflict in \( \text{action}[I_4, b] \), instead of \( S \rightarrow aa \). This was a typo on my part, and I will grade this question leniently.]
3. (15 points) Consider our usual bottom-up grammar for arithmetic expressions over constants and variables using “+” and “∗”:

\[
E' \rightarrow E \\
E \rightarrow E + T \mid T \\
T \rightarrow T * F \mid F \\
F \rightarrow c \mid v \mid (E)
\]

Add semantic rules to compute the following attributes:

- \(E'.v\) is the number of variable occurrences in the expression \(E'\). (Multiple occurrences of the same variable are counted).
- \(v.n\) is the nesting depth of each occurrence of \(v\) inside parentheses. For example, in the expression, \((v_1 + (c + v_2)) + v_3\), \(v_1\) has nesting depth 1, \(v_2\) has nesting depth 2, and \(v_3\) has nesting depth 0. (To be precise, the nesting depth of an occurrence of \(v\) is the difference of the number of opening parentheses to the left of the
occurrence minus the number of closing parentheses to the left of
the occurrence.)

You may define (within reason) any additional attributes that you
may find helpful, but your syntax-directed definition must be an L-
attributed definition.

**ANSWER:**

<table>
<thead>
<tr>
<th>production</th>
<th>rule(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E' \rightarrow E$</td>
<td>$E'.v := E.v; E.n := 0$</td>
</tr>
<tr>
<td>$E \rightarrow E_1 + T$</td>
<td>$E.v := E_1.v + T.v; E_1.n := E.n; T.n := E.n$</td>
</tr>
<tr>
<td>$E \rightarrow T$</td>
<td>$E.v := T.v; T.n := E.n$</td>
</tr>
<tr>
<td>$T \rightarrow T_1 * F$</td>
<td>$T.v := T_1.v + F.v; T_1.n := T.n; F.n := T.n$</td>
</tr>
<tr>
<td>$T \rightarrow F$</td>
<td>$T.v := F.v; F.n := T.n$</td>
</tr>
<tr>
<td>$F \rightarrow c$</td>
<td>$F.v := 0$</td>
</tr>
<tr>
<td>$F \rightarrow v$</td>
<td>$F.v := 1; v.n := F.n$</td>
</tr>
<tr>
<td>$F \rightarrow (E)$</td>
<td>$F.v := E.v; E.n := F.n + 1$</td>
</tr>
</tbody>
</table>

4. (10 points) Consider the following Pascal declaration:

```pascal
var
  a : array[1..9] of array[5..8] of Real;
```

Assuming that a Pascal Real is eight bytes and that the base address
of $a$ is 500, find the base address of the integer variable $a[6][7]$.

**ANSWER:** The size of an element of $a$ is $8 \cdot 4 = 32$ bytes. The
address is then

$$500 + (6 - 1)(32) + (7 - 5)(8) = 676.$$

5. (35 points total) Consider the following three-address code (line num-
bbers added):

```
1   L1:  i := a
2   L2:  if i <= 10 then goto L5
3       j := i + 1
4        goto L3
5   L3:  if j <= 10 then goto L5
6   L4:  i := j + 1
7        goto L2
```
8  L5:  a := a - 1
9    if a > 0 then goto L4
10
Assume that control enters at line 1 and that there are no other entry points.

(a) (5 points) Describe the basic blocks $B_1, B_2, \ldots$ by giving an inclusive range of line numbers for each block.

(b) (10 points) Draw the flow diagram as a directed graph, labeling the nodes $B_1, B_2, \ldots$. Give dangling arrows both for the entry point and for the exit point of the code as a whole.

(c) (5 points) Using the strict definition of a loop as defined in class and in the text, list any sets of vertices that constitute loops. Label any inner loop(s) as such.

(d) (15 points) Assuming $i$ is dead at the exit point (i.e., just before line 10), for each control point, say whether $i$ is alive or dead. Give your answer as a table, where each control point is given as “before line $n$,” for $n = 1, \ldots, 10$.

**ANSWER:**

(a)

<table>
<thead>
<tr>
<th>block</th>
<th>from line</th>
<th>through line</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_1$</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$B_2$</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$B_3$</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>$B_4$</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>$B_5$</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>$B_6$</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

(b) Entry block $B_1$ points to $B_2$: $B_2$ points to $B_3$ and $B_6$: $B_3$ points to $B_4$: $B_4$ points to $B_5$ and $B_6$: $B_5$ points to $B_2$: $B_6$ points to $B_5$ and the exit. ($B_3$ and $B_4$ could be merged.)

(c) There is one loop: \{ $B_2, B_3, B_4, B_5, B_6$ \}, which is an inner loop.

(d)
6. (10 points) Give an unambiguous grammar for the language of all strings of balanced delimiters of three types: parentheses, square brackets, and curly braces. Make your grammar as simple as possible.

**Answer:**

\[ S \rightarrow (S) | [S] | \{S\} | \varepsilon \]