1. (10 pts) Let $\Sigma = \{a, b\}$ be an alphabet and let $f: \Sigma^* \rightarrow N$ be a function from $\Sigma^*$ to nonnegative integer which counts the number of as in the input. $f$ can be defined inductively as follows:

Basis case: 0. If $x = \varepsilon$ then $f(x) = 0$.
Inductive case: 1. If $x = ax$, then $f(x) = 1 + f(y)$, and
2. If $x = by$, then $f(x) = f(y)$.

Now prove by structural induction (on $x$) that for all strings $x, y$, $f(xy) = f(x) + f(y)$.

The proof is by structural induction on $x$:

Basis case: $x = \varepsilon$, then $f(\varepsilon \cdot y) = f(y) = f(\varepsilon) + f(y) = f(\varepsilon) + f(y)$.

Inductive case: either $x = ax$ or $x = by$ for some $y \in \Sigma^*$.

Case 1: $x = ax$ implies $f(ax \cdot y) = f((ax) \cdot y) = f(a \cdot (x \cdot y)) = f(a) + f(x \cdot y) = f(x) + f(y)$.

Case 2: $x = by$ implies $f(by \cdot y) = f((by) \cdot y) = f(b \cdot (y \cdot y)) = f(b) + f(y)$.

2. (6 pts) Let $A$ and $B$ be two sets of strings. The what are the sets $AB$ and $A^*$?

(a) $AB = \{xy \mid x \in A \text{ and } y \in B\}$
(b) $A^* = \{\varepsilon\} \cup \bigcup_{k \geq 0} A^k = \{x_1 x_2 \cdots x_n \mid \forall k \geq 0, x_k \in A\}$

3. (10 pts) Let $A = \{aa, aaaa\}$ and $B = \{aaa, aaaaa\}$. Then which of the following statements are correct?

(a) $a^{101} \in A^*$.
(b) $a^{100} \in AB$.
(c) $\varepsilon \in A^*$.
(d) $A^* = AA^* \cup \{\varepsilon\}$.
(e) $(B^*)^* = B^*$

ANS: $a, d, e$ (1 point; 2 points for correct symbols, 3 points for context-free)

4. (39 pts) Given an NFA $M$ defined as follows:

<table>
<thead>
<tr>
<th>M</th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>1F</td>
<td>2</td>
<td>{1}</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>{}</td>
</tr>
<tr>
<td>3F</td>
<td>{}</td>
<td>{2,3}</td>
</tr>
</tbody>
</table>
(a) Draw a state transition diagram for the NFA M. [8 pts]

(b) Let $M = (Q, \Sigma, \delta, S, F)$. Then what are the contents of $Q$, $\Sigma$, $S$, and $F$, respectively. [7 pts]

\begin{align*}
Q &= \{1, 2, 3\} \\
S &= \{1, 2, 3\} \\
\Sigma &= \{a, b\} \\
F &= \{2, 3\}
\end{align*}

\begin{align*}
\delta &= \\
&= \\
&= \\
&= (1, a) \rightarrow 2 \\
&= (1, b) \rightarrow 1 \\
&= (2, a) \rightarrow 3 \\
&= (2, b) \rightarrow 3 \\
&= (3, a) \rightarrow 2 \\
&= (3, b) \rightarrow 2 \\
&= \frac{4}{\text{分}} \\
&= \frac{4}{\text{分}} \\
&= \frac{4}{\text{分}} \\
&= \frac{4}{\text{分}}
\end{align*}

(c) List 4 strings of different length accepted by the machine. [4 pts]

ANS: $\varepsilon, aab, aaba, b^* a (a b^*) b a b^*$

(d) Construct a DFA equivalent to the above NFA. Remember to indicate the set of old states each state of the new machine corresponds to. [8 pts]

(e) Find a regular expression equivalent to the FA. [7 pts]

$$b^* + b^* a (a b^* b)^* a b^*$$
(f) Write a strongly left linear grammar equivalent to the FA. [5 pts]

5. [10 pts] Let $A = \{a^{nxn} \mid n > 0 \}$ be the set of strings of as of length $n^2$. Use (game-theoretic) pumping lemma to show that $A$ is not regular.

6. [6 pts] Consider the grammar $G$:

$$S \rightarrow ST \mid a \quad T \rightarrow bS$$

Show that the input 'ababa' is a member of $L(G)$ by giving a derivation of 'ababa' from $S$.

Note that the derivation must be complete and is not shortcut.

7. (14 pts) CFG Design:

(a) Design a CFG to generate the set $\{a^nb^n \mid n \geq 0 \}$

(b) Design a CFG to generate the set $\{a^mb^n \mid m, n \geq 0 \text{ and } m \neq n \}$
8. (10 pts) Consider the following pair of machines:

<table>
<thead>
<tr>
<th>M1</th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3F</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>M2</th>
<th>a</th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>2F</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Use the product construction to construct a DFA that accepts the set \( L(M_1) \) \(-\) \( L(M_2) \).
Remember to mark initial and final states of the new machine.

9. (15 pts) Consider the context free grammar \( G \):

\[
S \rightarrow [S] \mid SS \mid \epsilon
\]

(a) Devise a grammar \( G_1 \) such that \( L(G_1) = L(G) - \{\epsilon\} \) and \( G_1 \) contains no empty rule.[7 pts]

(b) Design a grammar in Chomsky normal form which is equivalent to \( G_1 \).

\[
\begin{align*}
S &\rightarrow [S] \\
 &\rightarrow [L] \\
 &\rightarrow SS
\end{align*}
\]

\[
\begin{align*}
S &\rightarrow [D] \\
 &\rightarrow [D] \\
 &\rightarrow \epsilon
\end{align*}
\]

\[
\begin{align*}
S &\rightarrow [C][S] \\
 &\rightarrow [C][O][S] \\
 &\rightarrow [S][S][S][S][S]
\end{align*}
\]

\[
[O] \rightarrow [O]
\]

\[
\rightarrow [O][C][S]
\]

Note: [D] 可改用 A
[D] 可改用 B
[D] 代替 C
[D] 代替 D