(1) (a) The following PDA accepts $L_1 = \{ w \in \{0,1\}^* \mid w = 0^i 1^j \text{ such that } j = 2i \text{ or } i = 2j \}$ by empty stack.

(b) The following PDA accepts $L_2 = \{ w \in \{0,1\}^* \mid w \text{ contains more 1s than 0s} \}$ by final state.

(2) (a) $L(G) = \{ w \in \{0,1\}^* \mid w \text{ does not contain 011 as a substring} \}$.

(b) $S \Rightarrow T \Rightarrow 1$ and $S \Rightarrow 1S \Rightarrow 1T \Rightarrow 1$ are two leftmost derivations for $1 \in L(G)$ so $G$ is ambiguous.

(c) Consider the context-free grammar $G' = (\{S', T'\}, \{0,1\}, P', S')$ with productions $P'$ given below

\[
S \rightarrow 1S \mid 0T \mid \epsilon \\
T \rightarrow 0T \mid 10T \mid 1 \mid 0.
\]

(3) (a) The only nullable symbol is $S$. This leads to the following equivalent grammar

\[
S \rightarrow ASB \mid AB \\
A \rightarrow aAS \mid aA \mid a \\
B \rightarrow SbS \mid Sb \mid bS \mid b \mid A \mid bb.
\]
(b) The only unit pair is \((B, A)\). This leads to the following equivalent grammar

\[
S \rightarrow ASB \mid AB \\
A \rightarrow aAS \mid aA \mid a \\
B \rightarrow SbS \mid bS \mid b \mid bS \mid aAS \mid aA \mid a.
\]

(c) All symbols \(S, A, B\) are reachable. All symbols \(S, A, B\) are generating. Therefore, there are no useless symbols.

(d) Add new variables for each terminal to obtain the following equivalent grammar

\[
S \rightarrow ASB \mid AB \\
A \rightarrow TaAS \mid TaA \mid a \\
B \rightarrow STbS \mid STbS \mid Sb \mid bS \mid Sb \mid b \mid bS \mid aAS \mid aA \mid a. \\
Ta \rightarrow a \\
Tb \rightarrow b.
\]

"Daisy chain" the productions whose righthand sides contain at least three variables to obtain the following equivalent grammar

\[
S \rightarrow AX_1 \mid AB \\
X_1 \rightarrow SB \\
A \rightarrow TaX_2 \mid TaA \mid a \\
X_2 \rightarrow AS \\
B \rightarrow SX_3 \mid STbS \mid b \mid SbTb \mid TaAS \mid TaA \mid a \\
X_3 \rightarrow TbS \\
X_4 \rightarrow AS \\
Ta \rightarrow a \\
Tb \rightarrow b.
\]

This is an equivalent grammar in Chomsky Normal Form (and contains no useless symbols).

(4) (a) The CFG \(G_2 = (\{S, T\}, \{0, 1, 2\}, P, S)\) with productions \(P\) given below

\[
S \rightarrow 0S22 \mid T \\
T \rightarrow 1T2 \mid \epsilon
\]

has \(L(G) = \mathcal{L}_2 = \{w \in \{0, 1, 2\}^* \mid w = 0^i1^j2^k \text{ and } k = 2i + j\}\).

(b) Consider \(s = 0^p1^{2p}2^p\). Observe \(s \in \mathcal{L}_1 = \{w \in \{0, 1, 2\}^* \mid w = 0^i1^j2^k \} \text{ and } |s| > p\). Suppose \(u, v, w, x, y\) are chosen such that (1) \(s = uvwxxy\), (2) \(|vwx| \leq p\), and (3) \(|vx| > 0\). Let \(i = 0\) and consider \(uv^ix^jw^jz = uwy\).

Case one: \(v\) contains a 0. In this case \(v\) and \(x\) do not contain any 2s, so \(uwy \notin \mathcal{L}_1\). Case two: \(v\) does not contain a 0. In this case \(v\) and \(x\) do not contain any 0s, and they do contain at least one 1 or 2. Thus, \(uwy \notin \mathcal{L}_1\).
In both cases the string \( uwx \notin L_1 \). Therefore, \( L_1 \) does not satisfy the pumping lemma for context-free languages and hence is not a context-free language.

(5) (a) If \( L_1 \) and \( L_2 \) are CFLs, then there exist CFGs \( G_1 = (V_1, T_1, P_1, S_1) \) and \( G_2 = (V_2, T_2, P_2, S_2) \) such that \( L(G_1) = L_1 \) and \( L(G_2) = L_2 \). Consider the following CFG \( G = (V, T, P, S) \) where \( V = V_1 \cup V_2 \cup \{S\} \), \( T = T_1 \cup T_2 \), and \( P = P_1 \cup P_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\} \). Clearly, \( L(G) = L(G_1) \cup L(G_2) \) and so \( L_1 \cup L_2 \) is a CFL, Therefore, CFLs are closed under union.

(b) Suppose for the sake of contradiction that CFLs are closed under complements. This would imply that CFLs are closed under intersection since \( L_1 \cap L_2 = (L_1 \cup L_2)^c \). However, CFLs are not closed under intersection, so our supposition is false. In other words, CFLs are not necessarily closed under complements.

(c) The following language
\[
L_1 = \{ w \in \{0, 1, 2\}^* \mid w = 0^i1^j2^k \text{ and } i > j > k \}
\]
is not a CFL. Consider the language \( L_2 = 2^* \). Observe that the following language
\[
L_1L_2 = \{ w \in \{0, 1, 2\}^* \mid w = 0^i1^j2^k \text{ and } i > j \}
\]
is a CFL. In particular, \( L(G) = L_1L_2 \) for the following CFG \( G = (\{S, A, B\}, \{0, 1\}, P, S) \) with productions in \( P \) below
\[
\begin{align*}
S & \rightarrow AB \\
A & \rightarrow 0 \mid 0A \mid 0A1 \\
B & \rightarrow B2 \mid \epsilon.
\end{align*}
\]

(6) Observe that in a rightmost grammar, the start variable \( S \) only derives sentential forms with a at most one variable at the righthand side. That is, every derivation is of the form \( S \Rightarrow^* wX \) or \( S \Rightarrow^* w \) for some \( w \in T^* \) and \( X \in V \). The idea of this solution is to make a correspondence between this rightmost variable in a derivation and the current state in a computation of the automaton. We also make correspondences between productions of the form \( X \rightarrow aY \) in the grammar and transitions from state \( X \) to state \( Y \) that read in \( a \) in the automaton, as well as between the final states in the automaton and the nullable variables in the grammar.

(a) Choose the non-deterministic finite automaton \( M = (Q, T, \delta, i, F) \) as follows
\[
\begin{align*}
&Q = V \\
i &= S \\
&Y \in \delta(X, a) \iff X \rightarrow aY \text{ is a production in } P \\
&F = \{q \in V \mid q \rightarrow \epsilon \text{ is a production in } P\}
\end{align*}
\]
It can then be proven that \( L(M) = L(G) \).

(b) Choose the rightmost grammar \( G = (V, T, P, S) \) as follows
\[
\begin{align*}
&V = Q \\
P &= \{X \rightarrow aY \mid \delta(X, a) = Y\} \cup \{X \rightarrow \epsilon \mid X \in F\} \\
&S = i
\end{align*}
\]
It can then be proven that \( L(M) = L(G) \).