Problem 1

$$S \to TT \mid U$$
$$T \to 0T \mid T0 \mid \#$$
$$U \to 0U00 \mid \#$$

(a) $L(G) = \{ w \mid w \in \{0, \#\}^* \text{ and has exactly two } \#\text{'s} \}$

$\cup \{ 0^n \# 0^{2n} \mid n > 0 \}$

(b) $L(G)$ is not regular. Consider $s = 0^\ell \# 0^{2\ell}$ where ℓ is the critical length. If $s = xyz$ with $|y| > 0$, then y cannot contain the symbol $\#$, because pumping down will produce a string not in the language. But if y is entirely ~~to~~ the left or the right of $\#$, then pumping y will ~~produce~~ break the condition that 0's to the right are twice as many 0's to the left. Therefore, s cannot be pumped.

Problem 2

(a) $A = \{ a^m b^n c^n \mid m, n \geq 0 \}$

Here's a grammar for A

$$
\begin{aligned}
S &\to AT \\
A &\to aA \mid \varepsilon \\
T &\to bTc \mid \varepsilon
\end{aligned}
$$

This grammar generate as many a's as we want to the left, and an equal number of b's and c's to the right. We conclude that A is CFL.

$B = \{ a^n b^n c^m \mid m, n \geq 0 \}$

A similar grammar can be constructed for B, proving B is also CFL.

If $C = A \cap B$, then

$C = \{ a^n b^n c^n \mid n \geq 0 \}$

we know that C is not CFL, so context-free languages are not closed under intersection.

(b) Since $A \cap B = \overline{\bar{A} \cup \bar{B}}$, and context-free languages are closed under the union operation, then if CFL is closed under complement, it would be closed under intersection, a contradiction. So CFL is not closed under complement.

Problem 3.

Give a CFG for the following language.

(a) $\{ w \in \{a,b\}^*, \text{ where } a(w) - 2b(w) = 0 \}$

where $a(w)$ is the number of a's in w and $b(w)$ is defined similarly.

A simple solution is the following grammar:

$$
\begin{aligned}
S \rightarrow \varepsilon \;&|\; aSaSbS \\
&|\; aSbSaS \\
&|\; bSaSaS
\end{aligned}
$$

where each rule generates twice as many a's as b's (0 each or 2 a's and 1 b), with all possible placements.

a ... a ... b

a .... b .... a

b .... a ... a

However, we can come up with a simpler one:

$$
\begin{aligned}
S \rightarrow \varepsilon \;|\; SS \;&|\; aaSb \\
&|\; bSaa \\
&|\; aSbSa
\end{aligned}
$$

To prove this works, let $f(w) = a(w) - 2b(w)$. We need $f(w) = 0$ for every string in this language, which is obvious. We also need every w s.t. $f(w) = 0$ to be in the language. We consider ~~4~~ 5 cases.

- $w = \varepsilon$. The rule $S \to \varepsilon$ takes care of this.
- $w = uv$, where $f(u) = f(v) = 0$. The rule $S \to SS$ takes care of this
- $f(u) > 0$ for every proper prefix u of w. Therefore if $w = uv$, then $f(u) > 0$. This means w must start with aa and end with a b. ~~[illegible]~~ So ~~[illegible]~~ $w = aaxb$ where $f(x) = 0$. The rule $S \to aaSb$ takes care of this.
- $f(u) < 0$ for every proper prefix u of w. This is the symmetric case to above, and $w = bxaa$ where $f(x) = 0$. The rule $S \to bSaa$ takes care of this.
- $f(u) > 0$ for some u where $w = uv$, and $f(u') < 0$ for some u' where $w = u'v'$. In this case, $f(u)$ can only transition from positive to negative as we move right; otherwise, it will pass through 0 and we hit the second case. So w must start with a. The first b that brings f to negative must make f go from $+1$ to -1. So w has the form $w = azbx$ where $f(z) = 0$. Furthermore, x must end with a because otherwise f would be positive just before the last b, which means it must transition from negative to positive so $w = azbya$ where $f(z) = f(y) = 0$. The rule $S \to aSbSa$ takes care of this.

(b) The compliment of the language $\{a^n b^n \mid n \geq 0\}$

Here we should note that any string that does not belong to the language above must be of the form

$$a^n b \Sigma^* b^n \quad \text{or} \quad a^n \Sigma^* a b^n$$

So we basically allow the string to have the the form $a^n \ldots\ldots b^n$, but we break the pattern by following the a^n with a b or preceding the b^n with an a.

$$S \rightarrow aSb \mid bT \mid Ta$$
$$T \rightarrow aT \mid bT \mid \varepsilon$$

(c) $\{w\#x \mid w^R \text{ is a substring of } x\}$ $\Sigma = \{0,1\}$

$$S \rightarrow AT$$
$$A \rightarrow 0A0 \mid 1A1 \mid \#T$$
$$T \rightarrow 0T \mid 1T \mid \varepsilon$$

(d) $\{x_1 \# x_2 \# \ldots \# x_k \mid k \geq 1,$ for some i and j, $x_i = x_j^R\}$

$S \to LCR$

$L \to \varepsilon \mid T\#$

$R \to \varepsilon \mid \#T$

$C \to aCa \mid bCb \mid \#T\# \mid a \mid b \mid \varepsilon$

$T \to aT \mid bT \mid \#T \mid \varepsilon$

<u>Problem 4</u> Pumping Lemma. Show that languages are not context-free

(a) $\{0^n 1^n 0^n 1^n \mid n \geq 0\}$

This is in Sipser's book for the language

$\{ww : w \in \{0,1\}^*\}$

(b) $\{0^n \# 0^{2n} \# 0^{3n} \mid n \geq 0\}$

If ℓ is the critical length, consider

$$s = 0^{\ell} \# 0^{2\ell} \# 0^{3\ell} = uvxyz$$

Now neither v nor y could contain #; otherwise pumping will produce a string with more than two #'s. So both v and y must contain only 0's. This means that one of the 3 parts of s will not be covered. Therefore, pumping will break the relation $\ell, 2\ell, 3\ell$.

(c) $\{ w\#x \mid w \text{ is a substring of } x \}$ Assume $\Sigma = \{a, b\}$

Consider $s = a^l b^l \# a^l b^l$ where l is the critical length. If $s = uvxyz$ then we have three cases:

- v and y are both to the left of #. Then uv^2xy^2z will produce more characters to the left of # and thus the string will not be in the language
- v and y are both to the right of #. Then uxz will not be in the language.
- v and y straddle #. Then since $|vxy| \leq l$ v and y are both contained in the $b^l \# a^l$ portion of the string. So v contains only b's and y contains only a's. ~~[illegible]~~ At least one of them is not empty. If $|v| > 0$, pump up. If $|y| < 0$, pump down. If both are not empty, pump either up or down. In all cases we produce a string not in the language of the form

$$a^l b^i \# a^j b^l$$

where $i > l$ or $j < l$.

(d) $\{ x_1 \# x_2 \# \cdots \# x_k \mid k \geq 2$ and for some $i \neq j$, $x_i = x_j \}$

Assume $\Sigma = \{a, b\}$

Consider $s = a^l b^l \# a^l b^l = uvxyz$

If any of v or y contain $\#$, then pumping down will produce a string not in the language. Otherwise part (c) shows that there is no way to pump and keep the left part and right part equal.

Problem 5. Chomsky Normal Form

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

- First we add a new start symbol so that it won't appear on the right.

$\underline{S \to A}$

$A \to BAB \mid B \mid \varepsilon$

$B \to 00 \mid \varepsilon$

- Second, we remove ε rules. Removing $B \to \varepsilon$

$S \to A$

$A \to BAB \mid B \mid \underline{AB} \mid \underline{BA} \mid \underline{A} \mid \varepsilon$

$B \to 00$

Removing $A \to \varepsilon$.

$S \to A \mid \underline{\varepsilon}$

$A \to BAB \mid \underline{BB} \mid B \mid AB \mid BA \mid A$

$B \to 00$

We don't have to remove $S \to \varepsilon$ since S is start variable.

- Third, remove unit rules. Removing $S \to A$.

$$S \to BAB \mid BB \mid B \mid AB \mid BA \mid \varepsilon$$
$$A \to BAB \mid BB \mid B \mid AB \mid BA \mid A$$
$$B \to 00$$

Removing $A \to A$

$$S \to BAB \mid BB \mid B \mid AB \mid BA \mid \varepsilon$$
$$A \to BAB \mid BB \mid B \mid AB \mid BA$$
$$B \to 00$$

Removing $A \to B$

$$S \to BAB \mid BB \mid B \mid AB \mid BA \mid \varepsilon$$
$$A \to BAB \mid BB \mid AB \mid BA \mid \underline{00}$$
$$B \to 00$$

Removing $S \to B$

$$S \to BAB \mid BB \mid AB \mid BA \mid \underline{00} \mid \varepsilon$$
$$A \to BAB \mid BB \mid AB \mid BA \mid 00$$
$$B \to 00$$

- Replace long rule.

$$S \to \underline{BT} \mid BB \mid AB \mid BA \mid \underline{ZZ} \mid \varepsilon$$
$$A \to \underline{BT} \mid BB \mid AB \mid BA \mid \underline{ZZ}$$
$$B \to \underline{ZZ}$$
$$\underline{T \to AB}$$
$$\underline{Z \to 0}$$

Problem 6

Convert CFG to PDA.

$E \to E + T \mid T$

$T \to T \times F \mid F$

$F \to (E) \mid a$