1. (**30 points,** from Sipser, problem 2.6)
   Give context free grammars generating the following languages:

   (a) **(10 points)** $\{w \mid \exists n \geq 0.\ (w = \mathtt{a}^n\mathtt{b}^{\mathtt{2n}}) \vee (\mathtt{w} = \mathtt{a}^{\mathtt{3n}}\mathtt{b}^{\mathtt{n}})\}$

   (b) **(10 points)** The complement of $\{w \mid \exists n \geq 0.\ w = \mathtt{a}^n\mathtt{b}^n\}$.

   (c) **(10 points)** $\{x_1\#x_2\#\cdots x_k \mid$ each $x_i \in \{\mathtt{a},\mathtt{b}\}^*$, and for some i and j, $x_i = x_j^{\mathcal{R}}\}$.

   For parts (a) and (b), the alphabet is $\{\mathtt{a},\mathtt{b}\}$. For part (c), the alphabet is $\{\mathtt{a},\mathtt{b},\#\}$.

2. (**30 points**) Give a PDA for each language from question 1. You can just draw a transition diagram where edges are labeled as in Sipser.

3. **(20 points)** Sipser, problem 1.13.
   Let $G = (V, \Sigma, R, S)$ be the following grammar: $V = \{S, T, U\}$; $\Sigma = \{\mathtt{0}, \#\}$; and R is the set of rules:

$$\begin{array}{lcl} S & \rightarrow & TT \mid U \\ T & \rightarrow & \mathtt{0}T \mid T\mathtt{0} \mid \# \\ U & \rightarrow & \mathtt{0}U\mathtt{00} \mid \# \end{array}$$

   (a) (**10 points**) Describe $L(G)$ in English.

   (b) (**10 points**) Prove that $L(G)$ is not regular.

4. (**25 points**, from Sipser, problem 2.19)
   Let G be the CFG

$$\begin{array}{lcl} S & \rightarrow & \mathtt{a}S\mathtt{b} \mid \mathtt{b}Y \mid Y\mathtt{a} \\ Y & \rightarrow & \mathtt{b}Y \mid \mathtt{a}Y \mid \epsilon \end{array}$$

   (a) (**15 points**) Give a simple description of $L(G)$ in English. Give a short explanation of your description.

   (b) (**10 points**) Use your answer to part (a) to give a CFG for $\overline{L(G)}$, the complement of $L(G)$.

5. (**20 points**), Sipser, problem 2.25
   For any language A, let $SUFFIX(A) = \{v \mid \exists u.\ uv \in A\}$. Show that the class of context-free languages is closed under the $SUFFIX$ operation.

6. (**25 points**), Sipser, problem 2.27
   Let $G = (V, \Sigma, R, S)$ be the following grammar:

$$\begin{array}{rcl} \mathit{STMT} & \rightarrow & \mathit{ASSIGN} \mid \mathit{IfThen} \mid \mathit{IfThenElse} \\ \mathit{IfThen} & \rightarrow & \mathtt{if\ condition\ then}\ \mathit{STMT} \\ \mathit{IfThenElse} & \rightarrow & \mathtt{if\ condition\ then}\ \mathit{STMT}\ \mathtt{else}\ \mathit{Stmt} \\ \mathit{ASSIGN} & \rightarrow & \mathtt{a:=1} \\ \\ \Sigma & = & \{\mathtt{if},\ \mathtt{condition\ then},\ \mathtt{else},\ \mathtt{a:=1}\} \\ V & = & \{\mathit{STMT},\ \mathit{IfThen},\ \mathit{IfThenElse},\ \mathit{ASSIGN}\} \end{array}$$

   G is a natural-looking grammar for a fragment of a programming language, but G is ambiguous.

   (a) Show that G is ambiguous.

   (b) Give a new, unambiguous grammar for the same language.