1. Constructing Four Grammars (30 points)
For each of the following, construct context-free grammars that generate the given set of strings. If your grammar has more than one variable, we will ask you to write a sentence describing what sets of strings you expect each variable in your grammar to generate.

For example, if your grammar were:

\[
S \rightarrow E | O \\
E \rightarrow EE | CC \\
O \rightarrow EC \\
C \rightarrow 0 | 1
\]

We would expect you to say “\(E\) generates (non-empty) even length binary strings; \(O\) generates odd length binary strings; \(C\) generates binary strings of length one.”

(a) [10 Points] All binary strings that contain at least one 1 and at most two 0’s.

(b) [10 Points] \(\{1^m0^n1^{m+n} : m, n \geq 0\}\)

(c) [10 Points] All strings of the form \(x\#y\), with \(x, y \in \{0, 1\}^*\), such that either \(x\) is a subsequence of \(y^R\) (i.e., it is \(y^R\) with some characters possibly removed) or \(y\) is a subsequence of \(x^R\).

2. Set Up To Relate (15 points)
Let \(A\) be a set. Let \(R\) and \(S\) be transitive relations on \(A\).

(a) [7 Points] Is \(R \cup S\) necessarily transitive? Prove your answer.

(b) [8 Points] Is \(R \cap S\) necessarily transitive? Prove your answer.

3. Symmetry and Power (25 points)
(a) [5 Points] Suppose that \(R\) and \(S\) are symmetric binary relations on a set \(A\). Show that \(S \circ R\) is not necessarily symmetric.

(b) [10 Points] Let \(R\) and \(S\) be as above. Suppose we also know that \(S \circ R = R \circ S\). Prove that \(S \circ R\) is symmetric.

(c) [10 Points] Use induction to show that \(R^n\) is symmetric for all integers \(n \geq 1\). You can use without proof the fact that relation composition \((\circ)\) is associative. (In other words, \(R^n\) is the same relation no matter where you place parentheses in \(R \circ R \circ \cdots \circ R\).)
4. DFAs [Online] (30 points)
For each of the following, create a DFA that recognizes exactly the language given.

(a) [10 Points] The set of all binary strings that contain at least two 1’s or at most two 0’s.

(b) [10 Points] Binary strings where if we treat them as a binary number, that number is congruent to 2 modulo 5. For example, 00111 is in the language (because $7 \equiv 2 \mod 5$) but 011 is not.

(c) [10 Points] Consider a binary string $w$ of even length. Let $\text{odd}(w)$ be characters at odd position in $w$ and $\text{even}(w)$ be characters at even position in $w$. For example, if $w = 100110$ then $\text{odd}(w) = 101$ and $\text{even}(w) = 010$. Consider $\text{even}(w)$ and $\text{odd}(w)$ as binary numbers. Let $L$ be the language

$$L = \{ w \in \{0, 1\}^* : w \text{ has even length and } \text{odd}(w) > \text{even}(w) \}.$$ 

For example, 00101101 $\in L$ because 0110 > 0011 but 001101 $\notin L$ because 010 $\not\succ$ 011. Also, 110011 $\notin L$.

You must submit and check your answers to this question using https://grinch.cs.washington.edu/cse311/fsms. You also must submit documentation in Canvas. For each problem, include a screenshot of your submitted DFA along with, for each state $s$ of your machine, a description of which strings will take the DFA from the start state to $s$.

5. EXTRA CREDIT: Ambiguity (0 points)
Consider the following context-free grammar.

- $\langle \text{Stmt} \rangle \rightarrow \langle \text{Assign} \rangle | \langle \text{IfThen} \rangle | \langle \text{IfThenElse} \rangle | \langle \text{BeginEnd} \rangle$
- $\langle \text{IfThen} \rangle \rightarrow \text{if condition then } \langle \text{Stmt} \rangle$
- $\langle \text{IfThenElse} \rangle \rightarrow \text{if condition then } \langle \text{Stmt} \rangle \text{ else } \langle \text{Stmt} \rangle$
- $\langle \text{BeginEnd} \rangle \rightarrow \text{begin } \langle \text{StmtList} \rangle \text{ end}$
- $\langle \text{StmtList} \rangle \rightarrow \langle \text{StmtList} \rangle \langle \text{Stmt} \rangle | \langle \text{Stmt} \rangle$
- $\langle \text{Assign} \rangle \rightarrow a := 1$

This is a natural-looking grammar for part of a programming language, but unfortunately the grammar is “ambiguous” in the sense that it can be parsed in different ways (that have distinct meanings).

(a) [0 Points] Show an example of a string in the language that has two different parse trees that are meaningfully different (i.e., they represent programs that would behave differently when executed).

(b) [0 Points] Give two different grammars for this language that are both unambiguous but produce different parse trees from each other.