0. Conceptual Review
(a) Regular expression rules:
   Basis: $\epsilon, \emptyset, a$ for $a \in \Sigma$
   Recursive: If $A, B$ are regular expressions, $(A \cup B), AB,$ and $A^*$ are regular expressions.

(b) Types of functions:
   One-to-one/Injection: A function $f$ is one-to-one iff $\forall a \forall b ((f(a) = f(b) \rightarrow a = b)$
   Onto/Surjection: A function $f : A \rightarrow B$ is onto iff $\forall b \in B \exists a \in A (b = f(a))$
   Bijection: A function $f : A \rightarrow B$ is a bijection if it is one-on-one and onto.

1. Regular Expressions Warmup
Consider the following Regular Expression (RegEx):

$$1(45 \cup 54)^*1$$

List 5 strings accepted by the RegEx and 5 strings from $T := \{1, 4, 5\}^*$ rejected by the RegEx. Then, summarize this RegEx in your own words.
2. Context Free Grammars Warmup
Consider the following CFG which generates strings from the language $V := \{0, 1, 2, 3, 4\}^*$

$$
S \rightarrow 0X4 \\
X \rightarrow 1X3 \mid 2
$$

List 5 strings generated by the CFG and 5 strings from $V$ not generated by the CFG. Then, summarize this CFG in your own words.

3. Simplify the RegEx
Consider the following Regular Expression (RegEx):

$$0^*(0 \cup 1)^*((01) \cup (11) \cup (00))1^*(0 \cup 1)^*$$

List 3 strings accepted by the RegEx and 3 strings from $S := \{0, 1\}^*$ rejected by the RegEx. Then, summarize this RegEx in your own words and write a simpler RegEx that accepts exactly the same set of strings.
4. Constructing RegExs and CFGs
For each of the following, construct a regular expression and CFG for the specified language.
(a) Strings from the language $S := \{a\}^*$ with an even number of $a$’s.

(b) Strings from the language $S := \{a, b\}^*$ with an even number of $a$’s.

(c) Strings from the language $S := \{a, b\}^*$ with odd length.

(d) (Challenge) Strings from the language $S := \{a, b\}^*$ with an even number of $a$’s or an odd number of $b$’s.
5. Structural Induction: CFGs

Consider the following CFG:

\[ S \rightarrow SS \mid 0S1 \mid 1S0 \mid \epsilon \]

Prove that every string generated by this CFG has an equal number of 1's and 0's.

**Hint 1:** Start by converting this CFG to a recursively defined set.

**Hint 2:** You may wish to define the functions \( \#_0(x), \#_1(x) \) on a string \( x \).
6. Constructing DFAs
For each of the following, construct a DFA for the specified language.
(a) Strings from the language \( \Sigma := \{a\}^* \) with an even number of \( a \)'s.

(b) Strings from the language \( \Sigma = \{a, b\} \) with odd length.

(c) Strings from the language \( \Sigma = \{a, b\} \) with an even number of \( a \)'s or an odd number of \( b \)'s.