The aim of this assignment is to give you some practice with formal languages, FSAs, and regular expressions.

Your assignment must be typed to produce a PDF document a3.pdf (hand-written submissions are not acceptable). You may work on the assignment in groups of 1 or 2, and submit a single assignment for the entire group on MarkUs.

1. Let $x \in \mathbb{N}$. Prove that $\text{term}(x)$ (below) terminates. Hint: Trace the values of $x$ and $y$ for a few different inputs, then devise a loop invariant that helps prove termination.

```python
def term(x):
    y = x**3
    while y != 0:
        x = x - 1
        y = y - 3 * x * x - 3 * x - 1
```

2. Let $\Sigma = \{a, b\}$, $L_a = \{a^k \mid k \in \mathbb{N}\}$, $L_b = \{b^j \mid j \in \mathbb{N}\}$, and $L_2 = \{x \in \{a, b\}^* \mid |x| \text{ is even}\}$. Do not draw the machines below. Specify them with the quintuple $(Q, \Sigma, \delta, Q_0, F)$, where transition function $\delta$ is a table with symbols on the rows and states on the columns.

   (a) Construct DFA $M_a$ such that $L(M_a) = L_a$. Be sure to include all states, including dead states. Devise a state invariant for $M_a$ and use it to prove that $M_a$ accepts $L_a$.

   (b) Construct DFA $M_b$ such that $L(M_b) = L_b$. Be sure to include all states, including dead states. Devise a state invariant for $M_b$ and use it to prove that $M_b$ accepts $L_b$.

   (c) Construct DFA $M_2$ such that $L(M_2) = L_2$. Be sure to include all states, including dead states. Devise a state invariant for $M_2$ and use it to prove that $M_2$ accepts $L_2$.

   (d) Construct DFA $M_{ab}$ such that $L(M_{ab}) = L_a \cup L_b$. Use the product construction from week 9 slides. Explain how you can use the proofs for $M_a$ and $M_b$ to establish that $M_{ab}$ accepts $L_a \cup L_b$.

   (e) Construct DFA $M_{ab\text{ even}}$ such that $L(M_{ab\text{ even}}) = (L_a \cup L_b) \cap L_2$. Explain how you can use the proofs for the preceding machine to establish that $M_{ab\text{ even}}$ accepts $(L_a \cup L_b) \cap L_2$.

3. Let $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, $L_0 = \{x \in \Sigma^* \mid x \text{ represents a number equivalent to } 0 \text{ mod } 3 \text{ in base } 10\}$, $L_1 = \{x \in \Sigma^* \mid x \text{ represents a number equivalent to } 1 \text{ mod } 3 \text{ in base } 10\}$, and $L_2 = \{x \in \Sigma^* \mid x \text{ represents a number equivalent to } 2 \text{ mod } 3 \text{ in base } 10\}$. Construct $M_0$ that accepts $L_0$ and $\{\varepsilon\}$, $M_1$ that accepts $L_1$, and $M_2$ that accepts $L_2$, being sure that each machine has exactly 3 states. Do not draw your machines, specify them with a quintuple.

   Now use the structure of your machines to explain why $L_0 = \text{Rev}(L_0)$, $L_1 = \text{Rev}(L_1)$, and $L_2 = \text{Rev}(L_2)$.
4. Let $\mathcal{RE}$ be the set of regular expressions over the alphabet $\Sigma = \{0,1\}$. Use structural induction on $\mathcal{RE}$ to prove:

(a) \[ \forall r \in \mathcal{RE}, \exists r' \in \mathcal{RE}, \text{Rev}(L(r)) = L(r') \]

(b) Using the definition of Prefix($L$) in Exercise 11 at the end of Chapter 7:
\[ \forall r \in \mathcal{RE}, \exists r' \in \mathcal{RE}, \text{Prefix}(L(r)) = L(r') \]

(c) If $r \in \mathcal{RE}$ does not contain the Kleene star, then $|L(r)|$ is finite.

5. Let $\Sigma = \{a, b, c\}$ and $L_{RA} = \{x \in \Sigma^* \mid |x| = 4 \land x = x^R\}$. Prove that any DFA that accepts $L_{RA}$ has at least nine states. Hint: consider what happens if 2 distinct strings in $L_{RA}$ of length 2 drive the machine to the same state. Generalize your result: what can you say about a DFA that accepts $L_R = \{x \in \Sigma^* \mid x = x^R\}$?