Problem 1

Prove that the language
\[ A = \{ 0^n 1^n 0^n \mid n \in \mathbb{N} \} \]
is decidable. Give an implementation-level description of the Turing machine that demonstrates this (i.e. give a formal description or an informal diagram).

Problem 2

Prove that the language
\[ B_{TM} = \{ \langle M_1, M_2, w \rangle \mid M_1 \text{ and } M_2 \text{ are both TMs that accept string } w \} \]
is Turing-recognizable.

Problem 3

Prove that if \( C \) is decidable, then \( C \circ \{0,1\}^* \) is decidable.

Problem 4

Let \( f : \{0,1\}^* \rightarrow \{0,1\}^* \) be a function that flips all the bits of its input. For example,
\[ f(0110) = 1001. \]
Prove that the language
\[ D = \{ \langle M \rangle \mid M \text{ is a DFA that accepts } w \text{ if and only if } M \text{ accepts } f(w) \} \]
is decidable.

Problem 5

Prove that the language
\[ N_{PDA} = \{ \langle P \rangle \mid P \text{ is a PDA that does not use its stack on accepting paths} \} \]
is decidable.
Bonus Problem:

Let $L$ be a language. Prove that $L$ is Turing-recognizable if and only if there exists a decidable language $\hat{L}$ such that

$$L = \{ x \mid \text{there is a } y \text{ such that } \langle x, y \rangle \in \hat{L} \}.$$ 

Bonus Problem: The Old Problem 2

Describe why the language

$$E_{\text{TM}} = \{ \langle M_1, M_2 \rangle \mid M_1 \text{ and } M_2 \text{ are TMs and } L(M_1) = L(M_2) \}$$

is NOT Turing-recognizable.