1. (10 points) A return-tape Turing machine is a variant of a Turing machine whose transitions use the directions RIGHT and RETURN instead of RIGHT and LEFT. RIGHT moves the tape head one character to the right and RETURN returns the tape head to the beginning of the input. All other aspects of the formal definition of a Turing machine in this model remain the same.

Prove that any language recognized by a return-tape Turing machine is Turing-recognizable, using an ordinary Turing machine to simulate the return-tape Turing machine.

2. (10 points) Consider the following computational problems:

   \[ EQ_{DFA} = \{ \langle A, B \rangle \mid A \text{ and } B \text{ are DFAs and } L(A) = L(B) \} \]
   \[ SUB_{DFA} = \{ \langle A, B \rangle \mid A \text{ and } B \text{ are DFAs and } L(A) \subseteq L(B) \} \]
   \[ DISJ_{DFA} = \{ \langle A, B \rangle \mid A \text{ and } B \text{ are DFAs and } L(A) \cap L(B) = \emptyset \} \]

Prove that \( SUB_{DFA} \) and \( DISJ_{DFA} \) are each Turing-decidable.

   You may (and should) use high-level descriptions of any Turing machines you define. Make sure to provide both a machine definition and a proof of correctness.

3. (10 points) We have seen that \( A_{TM}, E_{TM}, EQ_{TM} \), etc. are all undecidable problems. Give an example of a nonempty computational problem about Turing machines that is decidable.

   In other words, fill in the blanks below:

   \[ \{ \langle M \rangle \mid M \text{ is a Turing machine and } \ldots \} \]

   and prove that the resulting language is Turing-decidable.