1. **Turing machines.** [[1] p. 188, no. 3.8 part b] Give pseudo-code describing how a Turing machine moves the read/write head and manipulates the tape in order to decide the language:

   \{w | w contains twice as many 0s as 1s\}

2. **Two-stack PDAs.**

   Consider a variant on PDAs that have two stacks instead of one, where the transition function can consider the current input symbol as well as the top symbols on the stacks, and can push/pop to/from either or both stacks.

   (a) Give a formal definition of such a two-stack PDA model.

   (b) Give a construction that shows that the following is true:

      If language \( L \) is accepted by two-stack PDA \( P \), \( L \) Turing-decidable.

   (c) Give a construction that shows that the following is true:

      If language \( L \) is Turing-decidable, there is a two-stack PDA \( P \) accepting \( L \)?

3. **Closure Properties of Turing-decidable and Turing-recognizable Languages.**

   (a) How would you show that the class of Turing-decidable languages is closed under a certain operation \( o \)?

   (b) Show that the class of Turing-decidable languages is closed under complementation. I.e. show that if a language \( A \subseteq \Sigma^* \) for some alphabet \( \Sigma \) is Turing-decidable, then the language \( \Sigma^* - A \) is Turing-decidable as well.

   (c) The class of Turing-recognizable languages is not closed under complementation. Explain why your proof only works for the class of Turing-decidable languages and not for the class of Turing-recognizable languages.

**REFERENCES**


**HONOR CODE AFFIRMATION**

*I affirm that I have carried out my academic endeavors with full academic honesty.*