Final

(June 15, 2000)

Note

This is a closed-book exam. Each problem accounts for 10 points, unless otherwise marked.

Problems

1. Draw the state diagram of a pushdown automaton that recognizes the set of strings over $\{a, b\}$ with twice as many a's as b's. Explain the intuition behind the automaton by showing how it accepts the input string $abaaab$. Is the automaton deterministic or nondeterministic?

2. Give the implementation-level description of a (single-tape) Turing machine that decides the language in Problem 1.

3. Show that the collection of Turing-recognizable languages is closed under the operations of union and concatenation. Note that a recognizer may never halt when it does not accept the input.

4. Prove that, for any countable set A, there cannot exist a (one-to-one) correspondence between A and 2^A (the power set of A).

5. Let $SUBSET_{\mathrm{DFA}} = \{\langle M, N\rangle \mid M \text{ and } N \text{ are DFAs and } L(M) \subseteq L(N)\}$. Show that $SUBSET_{\mathrm{DFA}}$ is decidable.

6. Show that if A is Turing-recognizable and $A \leq_m \overline{A}$, then A is decidable.

7. Let $ZERO_{\mathrm{TM}} = \{\langle M\rangle \mid M \text{ is a TM and } L(M) = \{0\}\}$. Show that $ZERO_{\mathrm{TM}}$ is undecidable. (Hint: reduction from A_{TM}.)

8. In the proof of the Cook-Levin theorem, which states that SAT is NP-complete, we used 2×3 windows of cells to formulate the constraint that the configuration of each row (except the first one) in the $n^k \times n^k$ tableau follows legally from the configuration of the preceding row. Why couldn't we use an entire row of cells directly?

9. Let $DOUBLE\_SAT = \{\langle\phi\rangle \mid \phi$ has at least two satisfying assignments$\}$. Prove that $DOUBLE\_SAT$ is NP-complete.

10. What's wrong with the following arguments?

    Consider an algorithm for SAT: "On input ϕ, try all possible assignments to the variables. Accept if any satisfies ϕ." This algorithm clearly requires exponential time. Thus SAT has exponential time complexity. Therefore SAT is not in P. Because SAT is in NP, it must be true that P is not equal to NP.

Appendix

- $E_{\text{DFA}} = \{\langle A\rangle \mid A$ is a DFA and $L(A) = \emptyset\}$. E_{DFA} is decidable.
- $A_{\text{TM}} = \{\langle M, w\rangle \mid M$ is a TM and M accepts $w\}$. A_{TM} is undecidable.
- $A \leq_m B$ is equivalent to $\overline{A} \leq_m \overline{B}$.
- $SAT = \{\langle\phi\rangle \mid \phi$ is a satisfiable Boolean formula$\}$. SAT is NP-complete.