Exercises and Problems in Computability and Complexity

Robert Robere

January 10, 2016

In what follows, D is the class of decidable languages and SD is the class of semi-decidable languages. If M is any object then $\langle M \rangle$ denotes the encoding of M.

1 Uncategorized

1. [6] Design a multi-tape Turing Machine that takes a string w over $\{0, 1, \#\}^*$ as input and accepts the string if and only if it is of the form $\{x\#y\#z \mid x + y = z\}$, where the binary strings x, y, z are interpreted as encodings of positive integers in the usual way. First give a high-level description of the algorithm and then construct the actual Turing Machine.

2. [6] Prove that for each Turing Machine that halts under all inputs there is an equivalent Turing Machine (accepting the same language) that never moves its head to the left of its starting position. Give a complete formal description of the simulation.

3. [6] Prove that the set $S = \{(i, j, k) \mid i, j, k \in \mathbb{N}\}$ is countable by providing a one-to-one mapping from the set of natural numbers.

4. [4] Let $PAL = \{yy^r \mid y \in \{0, 1\}^*\}$ be the set of all binary palindromes. Give a Turing Machine which decides PAL — first give a high-level description of the algorithm, and then construct the actual Turing Machine.

5. [4] Prove $\mathsf{NP} \subseteq \mathsf{D}$.

6. [4] Prove the following *certificate definition* for SD: A language $L \in \mathsf{SD}$ if and only if there is a binary relation $R \subseteq \Sigma^* \times \Sigma^*$ such that R is computable and such that for all $x \in \Sigma^*$ we have
$$x \in L \Leftrightarrow \exists y : R(x, y).$$

7. [4] Let $A_{TM} = \{(\langle M \rangle, w) \mid M \text{ is a TM and } w \in \mathcal{L}(M)\}$. Prove that $A_{TM} \in \mathsf{SD}$.

8. [4] Show that a language L is decidable if and only if its characteristic function is computable.

9. [2, 4] Show that D is closed under complementation, union, and intersection. Is the same true for P and NP?

10. [2, Problem 3.14] Show that D is additionally closed under concatenation, and Kleene star.

11. [2, 4] Show that SD is closed under union and intersection. Is it closed under complementation?

12. [2, Problem 3.15] Additionally prove that that SD is closed under concatenation and Kleene star.

13. [3] Let L_1, L_2 be decidable languages. Prove or disprove the following: for any language L such that $L_1 \subseteq L \subseteq L_2$, L is decidable.

14. [4] Prove that $L \in \mathsf{D}$ if and only if $L \in \mathsf{SD}$ and $L \in \mathsf{coSD}$.

15. [4] Let $L \subseteq \Sigma^*$. Prove that $L \in \mathsf{SD}$ if and only if L has an enumerator.

16. [8] Show that if L is recursively enumerable, then there is a TM M that enumerates L without ever repeating an element of L.

17. Prove that non-deterministic and deterministic Turing Machines are equivalent in power (that is, prove that each can simulate the other).

18. [4] Prove that A_{TM} is not decidable.

19. [4] Prove that $\{(\langle M \rangle, w) \mid M \text{ is a TM and } w \notin \mathcal{L}(M)\}$ is not in SD. How is this language related to $\overline{A_{TM}}$?

20. [3] (TM can simulate Random Access). Given $x \in \{0,1\}^*$ we let $int(x)$ denote the non-negative integer encoded by x. Let

$$L = \left\{ x\#y \mid |y| \geq int(x) > 0 \text{ and } y_{int(x)} = 1 \right\}.$$

Design a Turing Machine that decides L.

21. [3] Define a "Lazy" Turing Machine to be a Turing Machine wherein the head can "stay in place" instead of moving to the Left or Right. Formally, the transition function of a Lazy TM is of the form $\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{L, S, R\}$, where S means that the head must "stay in place". Give a complete definition of what it means for a Lazy Turing Machine to compute. Then show that Lazy Turing Machines and Turing Machines are equivalent in power.

22. [3] Let $\Sigma = \{0, 1, 2, \ldots, 9\}$. Let Π denote the infinite string of the digits appearing in π, so that $\Pi = 3141592\ldots$. Define $L = \{x \in \Sigma^* \mid x \text{ is a subsequence of } \Pi\}$. Show that L is decidable. (Hint: Consider the set S of digits that occur infinitely often in Π, and a point in Π such that only members in S occur after that point.

23. Design a Turing Machine which decides the language $\{\langle G\rangle \mid \text{G is a connected, undirected graph}\}$.

24. [2, Exercise 3.5] Examine the formal definition of a Turing Machine to answer the following questions, and explain your reasoning:

    (a) Can a TM ever write the blank symbol on its tape?

    (b) Can the tape alphabet Γ be the same as the input alphabet Σ?

    (c) Can a TM's read-write head ever be in the same location in two successive steps?

    (d) Can a TM contain just a single state?

25. [2, Problem 3.9] Say that a *write-once Turing Machine* is a single-tape TM that can alter each tape square at most once. Show that this variant TM model is equivalent to the ordinary TM model.

26. [2, Problem 3.10] A *Turing Machine with doubly-infinite tape* is identical to an ordinary TM *except* its tape is infinite to the left as well as to the right. Show that this type of Turing Machine can recognize the class of semi-decidable languages.

27. [2, Problem 3.11] A *Turing Machine with left reset* is identical to an ordinary TM *except* that the transition function has the form

$$\delta\colon Q \times \Gamma \to Q \times \Gamma \times \{R, \text{RESET}\}\,.$$

    If $\delta(q, a) = (r, b, \text{RESET})$, then if the machine is in state q reading a, the machine's head jumps to the left-hand end of the tape after it writes b. Show that TMs with left reset recognize the class of Turing-recognizable languages.

28. [2, Problem 3.17] Show that single-tape TMs that cannot write on the portion of the tape containing the input string can only recognize regular languages.

29. [2, Problem 3.18] Let $\sum_{i=0}^{n} c_i x^i$ be a polynomial with a root at $x = x_0$. Let c_{max} be the largest absolute value of any c_i. Show that

$$|x_0| < (n+1)\frac{c_{max}}{|c_1|}.$$

30. [2, Problem 3.19] Let A be the language containing only the single string s where

$$s = \begin{cases} 0 & \text{if God does not exist} \\ 1 & \text{if God does exist.} \end{cases}$$

    Is A decidable? Why or why not?

31. [1, Problem 1.5]Define a TM M to be *oblivious* if its head movements do not depend on the input but only on the input length. That is, M is oblivious if for every input $x \in \Sigma^*$ and $i \in \mathbb{N}$, the location of each of M's heads at the ith step of execution on input x is only a function of $|x|$ and i. Show that for every TM M there is an oblivious TM M' such that $\mathcal{L}(M) = \mathcal{L}(M')$ and M' is a decider if and only if M is a decider.

32. Prove Rice's Theorem.

33. [8, Problem 2.8.6] Simulate a k-tape Turing machine by a single tape Turing machine by representing the k strings "on top of each other" — that is, replace the alphabet Σ with the alphabet Σ^k. How efficient can you make your simulation?

34. [8, Problem 2.8.7] Suppose that we have a Turing Machine with a 2-dimensional tape. Formally, the transition function δ is now of the form

$$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R, U, D\}$$

where U, D correspond to the tape moving "up" and "down". Carefully define the transition function and the definition of "yields" in such a machin, and show that such a TM can be simulated by a regular (single tape) Turing Machine.

35. [8, Problem 2.8.17] Show that any language decided by a k-tape nondeterministic TM within time $f(n)$ can be decided by a 2-tape nondeterministic TM *also* within time $f(n)$.

36. Design a Turing Machine M which computes the successor function.

37. [9, Problem 4.2.5] For any given deterministic TM M, construct a new TM M' such that $\mathcal{L}(M) = \mathcal{L}(M')$ but when M' halts its tape is always empty.

38. [9, Problem 4.2.10] (In this problem we consider an extension of read-only deterministic Turing Machines which are allowed to erase inputs. These are surprisingly powerful!) A TM M is a *read/erase* TM if it can, at each move, only read an input symbol and/or erase it. In other words, $\Gamma = \Sigma \cup \{\sqcup\}$ and the transition function satisfies, for all $q \in Q$ and $a \in \Gamma$, $\delta(q, a) = (q', a, D)$ or $\delta(q, a) = (q', \sqcup, D)$ for some $q' \in Q, D \in \{L, R\}$. Show that there is a read/erase TM that accepts $L = \{a^n b^n c^n \mid n \in \mathbb{N}\}$ (note that this implies that pushdown automata cannot simulate these machines). Then show that there is a decidable language not accepted by any read/erase TM.

2 Decidability and Semi-decidability

1. For each of the following languages L, determine if L and $\overline{L}$ are in D *without* using Rice's theorem.

   (a) [4] $HB = \{\langle M \rangle \mid M$ is a TM and M halts on the empty string.$\}$

   (b) [4] $NE = \{\langle M \rangle \mid M$ is a TM and $\mathcal{L}(M) \neq \emptyset\}$.

   (c) [4] $T = \{\langle M \rangle \mid M$ is a TM and M halts on every input.$\}$.

   (d) [4] $INF = \{\langle M \rangle \mid M$ is a TM and $\mathcal{L}(M)$ is infinite$\}$.

   (e) [4] $ALL = \{\langle M \rangle \mid M$ is a TM and $\mathcal{L}(M) = \Sigma^*\}$.

   (f) [4] $EQ = \{(\langle M_1 \rangle, \langle M_2 \rangle) \mid M_1, M_2$ are TMs and $\mathcal{L}(M_1) = \mathcal{L}(M_2)\}$.

(g) [6] $\{(\langle M\rangle, w) \mid M$ is a TM and on input w M moves its read-write head to the left at some point$\}$.

(h) [6] $\{(\langle M\rangle, x) \mid M$ is the lexicographically first TM that accepts $x\}$.

(i) [6] $\{(\langle M\rangle, x) \mid M$ is the lexicographically first TM that accepts x within x steps$\}$.

(j) [7] $\{\langle M\rangle \mid M$ accepts at least one string w beginning with $111\}$.

(k) [5] $\{\langle M\rangle \mid M$ is a TM and for each even n there is a length-n string $x \in \mathcal{L}(M)\}$.

(l) [5] $\{\langle M\rangle \mid M$ is a TM and 3 divides $|\mathcal{L}(M)|\}$.

(m) [5] $\{(\langle M\rangle, \langle q\rangle, w) \mid M$ is a TM, q is a state of M and M reaches q during the execution of $w\}$.

(n) [3] $\{\langle M\rangle \mid M$ is a TM and for some natural i M accepts and rejects some strings of length $i\}$.

(o) [3] $\{\langle M\rangle \mid M$ is a TM and M accepts infinitely many palindromes$\}$.

2. [4] Show that the set

$$\{\langle Q\rangle \mid Q \text{ is a multivariate polynomial with integral coefficients and an integral root}\}$$

is semi-decidable.

3. [4] Prove that there is a decidable language L that is *not* in NP.

4. [7] Prove that any finite language over the alphabet $\{0, 1\}$ is decidable. (This problem illustrates the importance of considering *infinite* languages).

5. [5] Prove that a language L is decidable if and only if A is many-to-one reducible to the language 0^*1^*.

6. [2,5] Prove that L is decidable if and only if there is an enumerator for L that enumerates the strings of L in lexicographic order.

7. [3] Prove that there is no semi-decidable language L that contains only encodings of deciding Turing Machines and for every decidable language L' there is a deciding Turing Machine M accepting L' such that $\langle M\rangle \in L$. (Note that this strengthens the usual theorem that the set of deciding Turing Machines is not semi-decidable).

8. [8] (This is the Turing Machine version of the "s-m-n" theorem) Show that there is an algorithm which, given a string x and the description $\langle M\rangle$ of a TM M accepting the language $\{x\#y \mid (x, y) \in R\}$ (where R is a relation) cosntructs the description of a TM M_x accepting the language $\{y \mid (x, y) \in R\}$. What does this result intuitively mean?

9. [9, Problem 5.2.3] Assume that $A, B, C \subseteq \{0, 1\}^*$ are three pairwise-disjoint sets.

Also, assume that there exist three partial computable functions f_1, f_2, f_3 such that

$$f_1(x) = \begin{cases} 1 & \text{if } x \in A \cup B \\ 2 & \text{if } x \in C \\ \perp & \text{otherwise} \end{cases}$$

$$f_2(x) = \begin{cases} \perp & \text{if } x \in A \cup C \\ 0 & \text{otherwise} \end{cases}$$

$$f_3(x) = \begin{cases} \perp & \text{if } x \in B \cup C \\ 0 & \text{otherwise} \end{cases}$$

Prove that all three sets $A, B, C.$ are decidable. Then, using the TMs M_1, M_2, M_3 which compute f_1, f_2, f_3, respectively, construct TMs that compute the characteristic functions of A, B, C.

10. [9, Problem 5.2.4] Prove that every infinite r.e. set has an infinite subset that is recursive.

11. [9, Example 5.3.23] We say that an r.e. set S is *simple* if its complement $\overline{S}$ is infinite but does not contain any infinite r.e. subset. Show that a simple r.e. set exists. (Hint: Use diagonalization)

12. [9, Problem 5.3.10] Show that there exists a set A such that both A and $\overline{A}$ are infinite but neither has an infinite r.e. subset.

13. [9, Problem 5.4.1] Assume that $A \cup B = \Sigma^*$ and $A \cap B \neq \emptyset$. Show that if A, B are semidecidable then $A \leq_m A \cap B$.

References

[1] Sanjeev Arora and Boaz Barak. Computational Complexity: A Modern Approach.

[2] Michael Sipser. Introduction to the Theory of Computation. International Thompson Publishing.

[3] Charles Rackoff. Homework for CSC 363, Winter 2015.

[4] Stephen A. Cook. Lecture Notes on Computability and Complexity. Lecture 1, Winter 2016.

[5] Toni Pitassi. Homework for CSC 448/2405, Winter 2006.

[6] Toni Pitassi. Homework for CSC 364, Winter 2002.

[7] Toni Pitassi. Test 1 for CSC 364, Winter 2002.

[8] Christos H. Papadimitriou. Computational Complexity.

[9] Ding-Zhu Du, Ker-I Ko. Problem Solving in Automata, Languages, and Complexity.