Computational Complexity

Problem set #1

Due date: 26 Esfand 97 (via lms.iut.ac.ir)

Solutions to the problems on this set might be found on the Internet. It is not acceptable to copy such solutions. Write with your own notation and formulation. In designing Turing machines, it would be enough if you describe the idea behind your solution as precise as possible, instead of expressing its formal construction.

Solve at least 6 problems to get a complete mark.

Problem 1. Design a Turing machine M deciding the following languages

(a) $L = \{s_1\#s_2\#s_3 \ : \ s_1, s_2, s_3 \in \{0,1\}^*, s_3 \textit{ is binary addition of } s_1 \textit{ and } s_2 \ \}$.

(b) $L = \{a^i b^j c^k \mid i \times j = k \text{ and } i, j, k \geq 1\}$.

Problem 2. (i) The *unary* representation of a natural number n, denoted by $\llcorner n \lrcorner_1$, is an string of length n whose all digits are one. For instance, $\llcorner 7 \lrcorner_1 = 1111111$. Design a deterministic Turing machine M_1 such that for any natural number n, $M_1(\llcorner n \lrcorner_2) = \llcorner n \lrcorner_1$, when $\llcorner n \lrcorner_2$ stands for the binary expansion of n.

(ii) Design a polynomial time deterministic Turing machine M_2 deciding the language

$$L = \{\sigma \in \{0,1\}^* : |\sigma| \text{ is a prime}\}.$$

(iii) Design a deterministic Turing machine M_3 to perform the primality test, that is, deciding the language

$$\text{PRIMES} = \{\sigma \in \{0,1\}^* : \text{The natural number with binary representaion } \sigma \text{ is a prime}\}.$$

Calculate the runtime of M_3 (Hint: amalgamate parts (i) and (ii)).

Problem 3. Design a nondeterministic Turing machine on the unary alphabet $\Sigma = \{1\}$ deciding the language $L = \{\sigma \in \{1\}^* : |\sigma| \text{ is composite}\}$ (note that 1 is neither prime nor composite).

Problem 4. Define a TM M to be ***oblivious*** if its head movement does not depend on the input but only on the input length. That is, M is oblivious if for every input $x \in \{0,1\}^*$ 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 $T : \mathbb{N} \to \mathbb{N}$, if $L \in \mathbf{DTIME}(T(n))$ then there is an oblivious TM that decides L in time $O(T(n)^2)$.

Problem 5. Prove that PRIMES$\in NP$ without using the fact that PRIMES$\in P$.
Hint: A number n is prime iff for every prime factor q of $n-1$, there exists a number $a \in \{2, \ldots, n-1\}$ satisfying $a^{n-1} = 1 \pmod{n}$ but $a^{(n-1)/q} \neq 1 \pmod{n}$. Use this fact and induction to guess certificates for prime factors of $n-1$.

Problem 6. A language is called unary if every string in it is of the form 1^i (the string of i ones) for some $i > 0$. Show that if there exists an NP-complete unary language then $P = NP$.

Problem 7. In the MAX CUT problem, we are given an undirected graph G and an integer K and have to decide whether there is a subset of vertices S such that there are at least K edges that have one endpoint in S and one endpoint in $\overline{S}$. Prove that this problem is NP-complete.

Problem 8. Show that $NP = coNP$ iff 3SAT and TAUTOLOGY are polynomial-time reducible to one another. TAUTOLOGY is the language of all Boolean functions which are satisfied by every assignment, $\{\varphi : \forall x, \varphi(x) = 1\}$.

Problem 9(Extra Credit). Let me recall ***Halting Problem***. Define

$$H_{TM} = \{\langle M, x\rangle | M \text{ is a TM and } M \text{ halts on } x\},$$

that is, the function H_{TM} takes as input a pair α, x and outputs 1 if and only if the TM M_α represented by α halts on input x within a finite number of steps. We know that, Halting problem in undecidable. Using Halting problem, provide a proof for Gödel's incompleteness theorem.

Good Luck.