

CSC 2401F 2008, Assignment 1

Due: October 7 at start of class

1. (25 points)

(a) Consider the language $pad(L) = \{w1^m \mid w \in L \text{ and } m = |w|\}$.

Show that $pad(L) \in \mathcal{P}$ implies that $L \in \mathcal{P}$.

Note: The algorithm level of description on say page 146 of the Sipser text is the appropriate level of description. I do NOT want detailed Turing machine programs.

(b) Show that there exists a language L such that $pad(L) \in DSPACE(n)$ but $L \notin DSPACE(n)$.

(c) Conclude $DSPACE(n) \neq \mathcal{P}$

2. (30 points)

Problem 7.23 in the Sipser text.

3. (20 points) Let L be a NP-complete language and suppose L is verified by polynomial time predicate V and certificate length bound $q(n)$. That is, $L = \{w \mid \exists y \text{ with } |y| \leq q(|w|) \text{ and } V(w, y) \text{ is true}\}$ and we say that the (short) certificate y verifies $x \in L$.

(a) The (L, V, q) search problem is as follows: Given w , determine whether or not $w \in L$ and then if $w \in L$, find a short certificate y that verifies $w \in L$. Show that the (L, V, q) search problem polynomial time reduces (in the sense of Cook-Turing) to the decision problem L . Hence if $L \in P$ (resp. $L \in DTIME(n^{O(\log n)})$) then the (L, V, q) search problem can be solved in polynomial time (resp. $DTIME(n^{O(\log n)})$).

(b) Consider the following “shortest certificate” optimization problem related to the (L, V, q) search problem: Given input w , determine if $w \in L$ and if so, compute the “shortest” (i.e. lexicographically first) certificate y that verifies $w \in L$. Show that the shortest certificate optimization problem polynomial time reduces (again in the sense of Cook-Turing) to the decision problem. Hence if $L \in P$ then the shortest certificate optimization problem can be solved in polynomial time.

4. (40 points) The following problem concerns three related scheduling problems.
- (a) The one machine unit profit scheduling problem with deadlines is as follows: We are given a set of jobs $S = \{J_1, \dots, J_n\}$ with each job J characterized by a pair (p_i, d_i) where p_i (resp. d_i) is the positive integer processing time (resp. deadline) for job J_i . A feasible schedule of S assigns each job $J_i \in S$ to an integer starting time s_i such that no two scheduled jobs overlap and $s_i + p_i \leq d_i$. Show that we can determine in polynomial time if S can be feasibly scheduled.
- (b) The two machine unit profit scheduling problem with deadlines is the above scheduling problem where now a feasible schedule of S assigns each job $J_i \in S$ to an integer starting time s_i on machine $m_i \in \{1, 2\}$ such that no two scheduled jobs overlap on a machine and again $s_i + p_i \leq d_i$. Show that the two machine unit profit scheduling problem with deadlines is weakly NP-hard. That is,
- Show that if all integer parameters are bounded by a polynomial in n , then the problem is solvable in polynomial time. Hint: Use dynamic programming.
 - If say integer parameter values can be exponential in n , then the problem is (weakly) \mathcal{NP} -hard.
- (c) The one machine unit profit scheduling problem with deadlines and release times is the same as the one machine scheduling problem above but now in addition to the processing times and deadlines, every job has a release time r_i . That is, a job is now characterized by a triple (r_i, p_i, d_i) with p_i and d_i as above and r_i is the non negative integer release time. Now a feasible schedule must also insure that $r_i \leq s_i$ for every scheduled job. Show that it is NP-hard to determine if all jobs in S can be feasibly scheduled.