Harvard University
Computer Science 121
Professor Salil Vadhan

Final Examination — SOLUTIONS
Saturday, January 21, 2006

Time: Three hours. Solve ALL the problems. The points total to 150.

PROBLEM 0 (4+4+4 points)

For each of the following conditions, either state that no language satisfies it, or give an example of a language that satisfies it. No justifications are necessary in either case.
(A) L is decidable but not recognized by any PDA.
(B) L is Turing-recognizable but not decidable.
(C) L is in P but not in NP.

(A) $L = \{a^n b^n c^n : n \geq 0\}$.
(B) $A_{\mathsf{TM}} = \{\langle M, w\rangle : w \in L(M)\}$, where M is a Turing machine.
(C) No language satisfies this (because $\mathrm{P} \subseteq \mathrm{NP}$).

PROBLEM 1 (15 points)

Give the 4-tuple for a grammar that generates the following context-free language over the alphabet $\Sigma = \{a, b\}$:

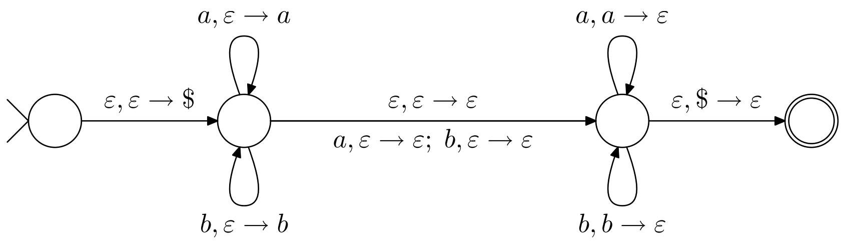
$$\{x \in \Sigma^* : x \neq x^R \text{ and } |x| \geq 1\}$$

The grammar generating the above language is $G = (V, \Sigma, R, S)$, with the rules R being:

$$\begin{aligned} S &\rightarrow aSa|bSb|aSb|bSa|T \\ T &\rightarrow aEb|bEa \\ E &\rightarrow aEa|bEb|a|b|\epsilon \end{aligned}$$

PROBLEM 2 (5+10 points)

(A) What language does the following PDA recognize?



(B) Prove that this language is not regular.

(A) The language recognized by the above PDA is $\{w \in \{a,b\}^* : w = w^R\}$.

(B) Suppose the language $L = \{w \in \{a,b\}^* : w = w^R\}$ is regular. By the pumping lemma, it has a pumping length of p. Consider the string $a^p b a^p \in L$. It can be partitioned into substrings x, y and z such that $xyz = a^p b a^p$ and $|xy| \le p$. This implies that $y = a^k$, for some $k \le p$. Pumping down, we get $xz = a^{p-k} b a^p \notin L$, since $a^{p-k} b a^p$ is clearly not a palindrome. This contradicts a condition of the pumping lemma. Therefore, L is not regular.

PROBLEM 3 (6+6+6+6+6+6 points)

Answer TRUE or FALSE. Justify your answers in a sentence or two.

(A) If L is regular, then $L' = \{x \in L : |x| > 2006\}$ is also regular.
(B) Every language is either Turing-recognizable or co-Turing-recognizable.
(C) If $f(n) = O(2^n)$, then $\mathrm{P} \subseteq \mathrm{TIME}(f(n))$
(D) Simulating a 2-tape TM by a 1-tape TM requires an exponential slowdown.
(E) $n^3 = o(n^3 + n^2)$.
(F) The number of languages *not* in NP is countable.

(A) TRUE. Let $L'' = \{x : |x| \le 2006\}$; this is regular because it is finite. Then $L' = L \cap \overline{L''}$ is regular because the class of regular languages is closed under intersection and complement.

(B) FALSE. There are only countably many TR languages (because each is described by a TM and there are only countably many TMs) and similarly there are only countably many co-TR languages. The union of two countable sets is countable, so there must be a language that is neither TR nor co-TR (because there are uncountably many languages). [Alternative justification: $\mathrm{EQ_{TM}} = \{\langle M_1, M_2\rangle : L(M_1) = L(M_2)\}$ is shown to be neither TR nor co-TR in Sipser.]

(C) FALSE. Take f to be the constant function $f(n) = 1$. Then $f(n) = O(2^n)$, yet P is not contained in $\mathrm{TIME}(f(n))$. [For example, $L = \{x \in \{a,b\}^* : \text{the last symbol of } x \text{ is } a\}$ is in P but cannot be decided in $O(1)$ time.]

(D) FALSE. In class and in Sipser, we saw that a time T computation on a multitape TM could be simulated in time $O(T^2)$ on a 1-tape TM. Simulating 1 step of the multitape TM required a constant number of scans across the simulated tape, which could not get bigger than $O(T)$ cells in a time T computation.

(E) FALSE. $\lim_{n\to\infty}[n^3/(n^3 + n^2)]$ equals 1 rather than 0 (as would be required for the statement to be true).

(F) FALSE. Every language in NP is Turing-recognizable and as mentioned in Part (B) above, there are only countably many TR languages. If there were also only countably many languages outside NP, then there would only be countably many languages altogether (as the union of two countable sets is countable), which we know to be false.

PROBLEM 4 (10 points)

List all languages recognized by DFAs having exactly two states, over the alphabet $\Sigma = \{a\}$.

- $\emptyset$
- $\{\varepsilon\}$
- $\{a^{2n} : n \in \mathbb{N}\}$
- $\{a^{2n+1} : n \in \mathbb{N}\}$
- $\{a^n : n \in \mathbb{N}\}$
- $\{a^n : n \geq 1\}$

PROBLEM 5 (15 points)

Complete the following table with `YES`, `NO`, or `?? (CURRENTLY UNKNOWN)`. No explanations needed. In the following table, M always stands for a Turing machine and R always stands for a regular expression.
(One point per box.)

Language	decidable	Turing-recognizable	co-Turing-recognizable	P	NP
$\{\langle R\rangle : R$ generates $abba\}$	YES	YES	YES	YES	YES
$\{\langle M\rangle : \lvert L(M)\rvert < 2006\}$	NO	NO	YES	NO	NO
$\{\langle M\rangle : M$ halts on all inputs$\}$	NO	NO	NO	NO	NO

Language	decidable	Turing-recognizable	co-Turing-recognizable	P	NP
$\{\langle R\rangle : R$ generates $abba\}$	YES	YES	YES	YES	YES
$\{\langle M\rangle : \lvert L(M)\rvert < 2006\}$	NO	NO	YES	NO	NO
$\{\langle M\rangle : M$ halts on all inputs$\}$	NO	NO	NO	NO	NO

PROBLEM 6 (10+15 points)

For each of the following languages, say whether they are in P, are NP-complete, or neither. Prove your answers.
(A) 3-CLIQUE= $\{G : G$ has a clique of size $3\}$.
(B) MULT3-CLIQUE= $\{(G, k) : G$ has a clique of size k, and k is a multiple of $3\}$.

(A) 3-CLIQUE is in P. Here is a polynomial-time decider for it:

M = On input $\langle G \rangle$:

1. For $i = 1$ to n (n is the number of vertices in G):
2.   For $j = 1$ to n:
3.   For $k = 1$ to n:
4.   If $i \neq j$ and $j \neq k$ and $k \neq i$:
5.   If $\{v_i.v_j, v_k\}$ forms a clique (v_l denotes the l-th vertex of G):
                       Accept;
6. Reject;

The decider simply enumerates all possible vertex subsets of size 3 (Step 1–4), and check if any of them forms a 3-Clique (Step 5). Both can be done in polynomial time and hence 3-CLIQUE is in P.

(B) MULT3-CLIQUE is NP-complete. To show that MULT3-CLIQUE is in NP, we use k-cliques as certificates. A polynomial verifier then simply does the following: On input $\langle G, k, S \rangle$, check that $|S| = k, k \equiv 0 \bmod 3$, and S is a clique. Clearly all of these can be done in polynomial time.

To show MULT3-CLIQUE is NP-hard, we reduce CLIQUE to MULT3-CLIQUE. Define a polynomial-time mapping f that maps $\langle G, k \rangle$ to $\langle G', k' \rangle$:

f = On input $\langle G, k \rangle$:

1. Let $G' = G$, $k' = k$.
2. While $k' \not\equiv 0 \bmod 3$:
3.   Add an additional vertex v to G' and connect v to all existing vertices in G'.
4.   Let $k' = k' + 1$.
5. Output $\langle G', k' \rangle$.

f is clearly polynomial-time. Moreover, if $\langle G, k \rangle \in$ CLIQUE, then there is a k-clique S in G. But G' is just G with $k' - k$ additional vertices, and these additional vertices are connected to each other and to all vertices in G. Hence S together with the additional vertices forms a k'-clique in G', i.e., $\langle G', k' \rangle \in$ MULT3-CLIQUE.

Conversely, suppose $\langle G', k' \rangle \in$ MULT3-CLIQUE, so there is a k'-clique S' in G'. Since the $k' - k$ additional vertices inserted in Step 3 are connected to each other and to other vertices in G, we can always add them to S' to form a new clique S'' of size $\geq k'$ (although it is possible that they are already in S', in which case $S' = S''$). But any subset of a clique is a also a clique. So if we remove the $k' - k$ additional vertices from S'', then we are left with a clique S of size $> k$. All vertices in S are in the original G. Therefore, $\langle G, k \rangle \in$ CLIQUE. Thus we have shown that f is a correct polynomial-time mapping reduction. It follows that CLIQUE $\leq_p$ MULT3-CLIQUE, and MULT3-CLIQUE is NP-hard since CLIQUE is NP-hard.

PROBLEM 7 (7+15 points)

Consider the "is polynomial-time reducible to" relation, denoted $\leq_p$, on the class of all languages in NP other than $\emptyset$ and Σ^*.
(A) In class, we showed that 3-SAT $\leq_p$ VERTEX COVER. Does VERTEX COVER $\leq_p$ 3-SAT? Justify your answer.
(B) Is $\leq_p$ reflexive? symmetric? transitive? If the answer to any of these properties is "uncertain," give an assumption that would make the property hold and an assumption that would make the property not hold. Justify your answers. (Remember that we explicitly excluded $\emptyset$ and Σ^* from the relation's domain.)

(A) Since VERTEXCOVER and 3SAT are both NP-complete, both are in NP, and all languages in NP leq_p-reduce to both of them. Therefore VERTEXCOVER leq_p 3SAT and 3SAT leq_p VERTEXCOVER

(B)

- The relation is reflexive. If L is a language, the identity function is a reduction from L to L. The identity function easily runs in polnomial time.
- The relation is transitive. Let $A \leq_p B$ by the reducing function f, which runs in polynomial time $p(n)$, and let $B \leq_p C$ by the reducing fucntion g, which runs in polynomial time $q(n)$. Then the function $g \cdot f$ is a reduction from A to C. It runs in time $q(p(n))$, which is polynomial since the polynomials are closed under composition.
- The relation is symmetric if and only if P = NP. First, if P = NP, then all languages in P except $\emptyset$ and Σ^* are NP-complete and thus reduce to eachother. If P $\neq$ NP, then 3SAT $\not\leq_p$ $\{a\}$, since $\{a\}$ is known to be in P.