Theory of Computation

Final Exam. Fall 2008

Name: ____________________ Student ID ________________ Phone No. ________________

(第 **1, 3** 題作答在題目卷上; 其他作答在答案卷上)

1. (70 pts) True or False. (Mark **O** for true, × for false. $\boxed{\text{Score = max\{Right – 1/2 Wrong\}}}$. (We have the following abbreviations: *FA≡finite automata; PDA≡pushdown automata, TM≡Turing machines; SD≡semi-decidable; D≡decidable*)

1. _O_ $a^*b^*c^*$ - $\{a^nb^nc^n, n \ge 0\}$ is context-free but not regular.
2. _O_ $\{w \in \{a, b\}^*$: the first, middle, and last characters of w are identical$\}$ is context-free.
3. X_ Define middle(L) = $\{x: \exists y, z \in \Sigma^* (yxz \in L)\}$. Then middle($\{a^nb^nc^n, n \ge 0\}$) = $a^*b^*c^*$.
4. _O_ Context-free languages are closed under $pref(L) = \{w: \exists x \in \Sigma^* (wx \in L)\}$.
5. _O_ Context-free languages are closed under $middle(L) = \{x: \exists y, z \in \Sigma^* (yxz \in L)\}$.
6. _O_ $\{a^nb^n, n \ge 0\} \cup \{a^nc^n, n \ge 0\}$ is deterministic context-free.
7. _X_ $\{a^nb^n, n \ge 0\} \cup \{a^nb^{2n}, n \ge 0\}$ is deterministic context-free.
8. _X_ If $L = L_1 \cap L_2$ and L and L_2 are context-free, then L_1 must be context-free.
9. _X_ If $L = L_1L_2$ and L is context-free, then L_1 must be context-free.
10. _X_ *If* L is context-free then $L \cap L^R$ is also context-free.
11. _O_Context-free languages are not closed under $shuffle(L) = \{w : \exists x \in L$ (w is some permutation of x)$\}$.
12. _X_If L is context free and R is regular, R - L must be context-free.
13. _X_A context-free grammar in Chomsky normal form is always unambiguous.
14. _X_If $L = L_1^+$ and L is context-free, then L_1 must be context-free
15. _O_Context-free languages are closed under concatenation with regular languages.
16. _O_ Context-free languages over a single character alphabet (i.e., $\Sigma=\{a\}$) are closed under intersection.
17. _O_The language $L = \{<A, B> : A$ and B are finite automata and $L(A) \subseteq L(B)\}$ is D.
18. _X_$\{<M, q>$: there is some reachable configuration (p, $u\underline{a}v$) of TM M, with $p \ne q$, that yields a configuration whose state is $q\}$ is D.
19. _O_$\{<M, w> : M$, on input w, never writes on its tape$\}$ is D.
20. _O_$\{\langle M\rangle : L(M)$ contains at least one odd length string$\}$ is SD but not D. .
21. _X_$\{\langle M\rangle : |L(M)| > 3\}$ is not semi-decidable, where $|L(M)|$ denotes the number of elements in $L(M)$.
22. _X_$\{<M, q> : M$ never enters state q when started on an empty tape$\}$ is SD but not D.
23. _O_$\{\langle M\rangle : |L(M)| \le 3\}$ is not SD.
24. _O_$\{\langle M\rangle : M$ fails to accept at least one of aabb or bbaa$\}$ is not SD
25. _O_2-way PDA (i.e., PDA whose input heads can move both left and right) are more powerful than 1-way PDA.
26. _O_Deterministic 2-counter machines are Turing equivalent.
27. _O_Given a PDA M_1 and an FA M_2, the problem "$L(M_1) \subseteq L(M_2)$?" is D.

28. O_ Given a TM M_1 and an FA M_2, the problem "$L(M_1) \cap L(M_2) \neq \varnothing$?" is SD.
29. X_ Given a TM M_1 and a PDA M_2, the problem "$L(M_1) \cap L(M_2) = \varnothing$?" is SD.
30. _X_ Given a TM M_1 and an FA M_2, the problem "$L(M_2) \subseteq L(M_1)$?" is SD.
31. _X_ The Church-Turing Thesis says that the halting problem for Turing machines is undecidable.
32. _X_ If L_1 and L_2 are SD, so is $L_1 - L_2$.
33. _O_ If L_1 and L_2 are D, so is $L_1 - L_2$
34. _O_ {<*M*> : *L*(*M*) is uncountably infinite} is D.
35. O_ Given a TM M, the problem of deciding whether M ever reads a blank symbol is D.

2. (6 pts) Use Ogden's Lemma to prove that the following is not context-free

$\{a^i b^j c^k d^n : i \geq 0, j \geq 0, k \geq 0, n \geq 0, \text{ and } i = 0 \text{ or } j = k = n\}$.

(Proof) Let k be the constant from Ogden's Lemma. Let $w = a^k b^k c^k d^k$. Mark all and only the b's in w as distinguished. If either v or y contains more than one distinct symbol, pump in once. The resulting string will not be in L because it will have letters out of order. At least one of v and y must be in the b region. Pump in once. The resulting string will still have a non-zero number of a's. Its number of b's will have increased and at most one of the c's and d's can have increased. So there are no longer equal numbers of b's, c's, and d's. So the resulting string is not in L.

3. (14 pts) Given two languages X and Y, we write $X \leq_m Y$ to denote that there is a mapping reduction (many-one reduction) from X to Y. Suppose that there are four languages *A*, *B*, *C*, and *D*. Each of the languages may or may not be semi-decidable. However, we know the following :

$$A \leq_m B, \quad B \leq_m C \quad \text{and} \quad D \leq_m C.$$

For each of the following statements, indicate whether it is:

- CERTAIN to be true, regardless of what the languages *A* through *D* are,
- MAYBE true, depending on what *A* through *D* are, or
- NEVER true, regardless of what *A* through *D* are.

NEVER_ A is semi-decidable but not decidable and C is decidable.

MAYBE_ A is not decidable and D is not semi-decidable.

CERTAIN If C is decidable, then the complement of D is also decidable.

NEVER The complement of A is not semi-decidable but the complement of B is.

NEVER The complement of B is not decidable but the complement of C is decidable.

MAYBE If A is decidable, then the complement of B is too.

MAYBE A is decidable but D is not.

4. (5 pts) Let $L_1, L_2, \ldots, L_k$ be a collection of languages over some alphabet Σ such that:

- For all $i \neq j$, $L_i \cap L_j = \varnothing$.
- $L_1 \cup L_2 \cup \ldots \cup L_k = \Sigma^*$.
- $\forall i$ (L_i is in SD).

Prove that each of the languages L_1 through L_k is in D.

Proof. $\forall i\ (\neg L_i = L_1 \cup L_2 \cup \ldots \cup L_{i-1} \cup L_{i+1} \cup \ldots \cup L_k)$.
Each of these L_j's is in SD, so the union of all of them is in SD. Since L_i is in SD and so is its complement, it is in D.

5. (5 pts) Prove that $\{<M> : M$ is a Turing machine and $|L(M)| = 12\}$ is not decidable. (Hint: Reduce from the Halting Problem. Do not use Rice's theorem.)

Proof.
Given <M,w>, construct the following M'

(1) Simulate M on w
(2) If halts, start a TM P that accepts a language of exactly 12 elements

See below

Clearly M halts on w iff |L(M')| = 12.