	Graduate Course
THEORY OF COMPUTATION	**College of Computer Science**
Final Exam	**Zhejiang University**
	Fall 2004

Class: ____________ **ID:** ____________ **Name:** ____________ **Score:**____________

Time allowed: 2 Hours.

1. **(20%) Determine whether the following statements are true or false. If it is true write a ✓ otherwise a × in the bracket before the statement.**
   (a) (   ) Every context-free language is recursive.
   (b) (   ) Language $\{a^m b^n c^l \mid m, n, l \in \mathbb{N}, m + n > 3l\}$ is context free.
   (c) (   ) All languages on an alphabet are recursively enumerable.
   (d) (   ) There's a language L such that L is undecidable, yet L and its complement are both semi-decided by the some Turing machine.
   (e) (   ) There's a function φ such that φ can be computed by some Turing machines, yet φ is not a primitive recursive function.
   (f) (   ) Let $L_1, L_2 \subseteq \Sigma^*$ be languages, recursive function τ is a reduction from L_1 to L_2, if L_1 is decidable, then so is L_2.
   (g) (   ) A language L is recursive if and only if it is Turing-enumerable.
   (h) (   ) Every language in $\mathcal{NP}$ is recursive.
   (i) (   ) Suppose A, B are two languages and there is a polynomial-time reductions from A to B. If A is $\mathcal{NP}$-complete, then B is $\mathcal{NP}$-complete.
   (j) (   ) Every language in $\mathcal{NP}$-complete can be reducible to the 3-SAT problem in polynomial time.

2. **(20%) FA and regular languages:**
   (a) Decide whether the following language is regular or not and provide a formal proof for your answer.

$$L = \{a^m b^n \mid m, n \in \mathbb{N}, (m - n) \mod 3 \neq 0\}$$

(b) Let Σ be an alphabet and let $L_1, L_2 \subseteq \Sigma^*$ be languages so that L_1 is not regular but L_2 is regular. Assume $L_1 \cap L_2$ is finite. Prove that $L_1 \cup L_2$ is not regular.

3. **(20%) PDA and Context-free languages:**

(a) Give a context-free grammar for the language

$$L_3 = \{xy \mid x, y \in \{a,b\}^*, |x| = |y| \text{ and } x \text{ and } y^R \text{ differ in one positions } \}.$$

For example, $abbbbaba, abbbbbbb \in L_3$, but $aababb \notin L_3$.

(b) Design a PDA $M = (K, \Sigma, \Gamma, \Delta, s, F)$ accepting the language L_3.

Solution: (b) The PDA $M = (K, \Sigma, \Gamma, \Delta, s, F)$ is defined below:

$K =$ _____

$\Sigma = \{a, b\}$

$\Gamma =$ _____

$s =$ _____

$F =$ _____

(q, σ, β)	(p, γ)

4. **(10%) Numerical Functions:**
Let $P(x, y)$ be primitive recursive predicates. Prove the following predicate

$$\forall y_{\leq u} P(x, y)$$

is also primitive recursive.

5. **(10%) Turing Machines**

Design a Turing machine for computing the following function.

$$f(x, y) = \begin{cases} 2x + 1, & \text{if } y \text{ is even} \\ 4x, & \text{if } y \text{ is odd} \end{cases}$$

where x and y are represented by binary strings respectively and separated with the symbol ";", i.e. the initial configuration in form of $\triangleright\underline{\sqcup}x; y$.

6. **(10%) Decidability and Undecidability**

   Show that the following language

   $$\{\text{“}M\text{”}\,\text{“}w\text{”} \mid M \text{ is a TM and } M \text{ halts on } w\}$$

   is recursively enumerable. An informal description suffices.

7. **(10%)$\mathcal{P}$ and $\mathcal{NP}$ Problems**

   Given n natural numbers $x_1, x_2, \cdots, x_n$ to test whether there exist distinct $i_1, i_2, \cdots, i_k$ such that $x_{i_1} + x_{i_2} + \cdots + x_{i_k} = (x_1 + x_2 + \cdots + x_n)/2$, and $x_{i_1} + x_{i_2} + \cdots + x_{i_k}$ is not a prime number. Design a $\mathcal{NP}$ algorithm for it, and estimate its time complexity.