NAME: ____

ROLL NO: ____

CS208 Mid-semester Exam (Spring 2013)

Time: 120 mins **Date: Feb 19, 2012**

- ***The exam is open-book, open-notes, open-material-brought-to-exam0hall.***
- ***Be brief, complete and stick to what has been asked. If needed, you may cite results/proofs covered in class without reproducing them.***
- ***You must write your answer neatly in only the space assigned for the answer.***
- ***For guessworks without any substantive justification, there will be negative 10 marks per question on which such guesswork is noticed.***
- ***Do not copy solutions from others***
- ***Penalty for copying: FR grade***

1. *[10 marks]* Let $\Sigma = \{0, 1\}$ and consider the language L_1 of words in Σ^* satisfying the following two properties:

   - Every word in L_1 has an equal number of 0's and 1's.
   - $L_1 \cap L(\mathbf{\Sigma}^* \cdot (\mathbf{0} \cdot \mathbf{0} \cdot \mathbf{0} + \mathbf{1} \cdot \mathbf{1} \cdot \mathbf{1}) \cdot \mathbf{\Sigma}^*) = \emptyset$.

   Is L_1 regular? If so, give an NFA of no more than 7 states accepting L_1. Otherwise, prove (using any technique that you have learnt in class) that L_1 is not regular.

2. *[10 marks]* In the following, u is said to be a sub-word of v iff there exist words x and z (possibly ε) such that $v = x \cdot u \cdot z$. Thus 01 and 001 are sub-words of 001100, but 010 is not a sub-word of 001100.

   In preparing for CS208 mid-semester exam, a student has come up with the following sequence of arguments.

   (a) Let $\Sigma = \{0, 1\}$ and define a homomorphism $h : \Sigma \to \Sigma^*$ as follows: $h(0) = 01$ and $h(1) = 10$.

   (b) Given words $u, w \in \Sigma^*$, define $N_u(w)$ to be the number of occurrences of u as a sub-word of w. Thus, $N_{00}(001100) = 2$ and $N_{00}(0000) = 3$. As discussed in class, the language $L = \{w \mid w \in \Sigma^*,\ N_{01}(w) = N_{10}(w)\}$ is known to be regular.

   (c) The inverse homomorphic image of L, i.e. $\widetilde{h}^{-1}(L)$ using notation introduced in class, is $\{v \mid v \in \Sigma^*,\ N_0(v) = N_1(v)\}$.

   (d) As discussed in class, regular languages are closed under inverse homomorphisms. Hence $\{v \mid v \in \Sigma^*,\ N_0(v) = N_1(v)\}$ is regular.

   If you think the student's arguments are correct, give an NFA with no more than 7 states to recognize the language $\{v \mid v \in \Sigma^*,\ N_0(v) = N_1(v)\}$. Otherwise, indicate which step(s) in the student's arguments are wrong and why.

3. *[10 + 10 marks]* For each of the following languages, construct an NFA (possibly with ε-transitions) that recognizes the language.

   (a) $L_1 = \{0^n 1^m 0^k \mid n, m, k \geq 0, \text{ and } (n - m) \bmod 2 = k \bmod 2\}$. For purposes of this problem, if $r \bmod 2 > 0$, consider $-r \bmod 2$ to be the same as $2 - (r \bmod 2)$. Otherwise (i.e. if $r \bmod 2 = 0$), consider $-r \bmod 2$ to be 0.

(b) $L_2 = \{0^n \mid n > 0, n^3 + n^2 + n + 1 = 0 \bmod 3\}$.

4. *[10 marks]* We have seen in class how to write a monadic second order logic (MSO) formula φ with the following properties to encode a regular set of words over any finite alphabet Σ:

   - The universe or domain over which φ is to be interpretted is the natural numbers $\mathbf{N}$ with the usual $<$ ordering on $\mathbf{N}$. In other words, all first order variables in φ take values from $\mathbf{N}$, and all second order variables in φ take values from subsets of $\mathbf{N}$.
   - Only monadic or unary predicates over $\mathbf{N}$ can be used in φ.
   - Both first order and second order variables in φ can be quantified using $\exists$ and $\forall$.
   - The usual Boolean operators $\wedge$, $\vee$ and $\neg$ can be applied on any sub-formula of φ.
   - There are special unary predicates $\mathsf{First} : \mathbf{N} \rightarrow \{\mathsf{True}, \mathsf{False}\}$ and $\mathsf{Last} : \mathbf{N} \rightarrow \{\mathsf{True}, \mathsf{False}\}$ such that $\mathsf{First}(x)$ evaluates to True only on the natural number 0, and $\mathsf{Last}(x)$ evaluates to True on a unique natural number > 0.
   - For every letter $a \in \Sigma$, we have a unary predicate $\mathsf{P}_a : \mathbf{N} \rightarrow \{\mathsf{True}, \mathsf{False}\}$ such that no two of these predicates evaluate to True on the same natural number.

   To simplify matters, suppose $\Sigma = \{a\}$. Consider the regular language $L = \{a^n \mid n \neq 1 \bmod 3\}$. Write a MSO formula φ such that every interpretation of the predicates in φ that causes φ to evaluate to True encodes a word in L and vice versa. Note that since $\Sigma = \{a\}$, all positions in a word must have the letter a. So we really need not have a monadic predicate called P_a. You are therefore free to use a MSO formula without the monadic predicate P_a (but with First and Last, of course). You must briefly justify the meaning of each sub-formula in your formula. Answers without justification will fetch no marks.