University of Waterloo
Midterm Examination

Spring 2013

Student Name _____

Student ID Number _____

Course Abbreviation & Number	CS 360
Course Title	Theory of Computation
Section	01
Instructor	dan brown

Date of Exam	June 3	
Time Period: in-class	Start Time: 12:30 pm	End Time: 1:20 pm
Duration of Exam:	50 minutes	
Number of Exam Pages (including this cover sheet)	6 pages	

1. Complete all answers in the spaces provided.
2. Write neatly so you do not lose marks unnecessarily.
3. Proctors will only confirm or deny the existence of errors on the exam.
4. In the case of perceived ambiguity, state a clear assumption and proceed to answer the question.
5. Cheating is an academic offence. Your signature on this exam indicates that you understand and agree to the university's policies regarding cheating on exams.

#	Marks	Actual	Initial
1	8		
2	12		
3	12		
$\sum$	32		

Signature:_____

1. (8 marks): **Recursive definitions and induction**
   Recall that L is a regular language if $L = \emptyset, L = \{\varepsilon\}, L = \{a\}$, where $a \in \Sigma$, or if $L = L_1 \cup L_2$ or $L = L_1 L_2$ for regular languages L_1 and L_2, or if $L = L_1^*$ for regular language L_1.

   (a) (4 marks) Show that if L is regular, then so is L^k for every nonnegative integer k.

   (b) (4 marks) Given that L^k is regular for every regular language L and non-negative integer k, do we still need the rule that L^* is also regular for every regular language L? Explain why or why not.

2. (12 marks): **Regular languages**

   (a) (6 marks) Consider the language $L_1 = \{w \in \{0,1\}^* \mid 010 \text{ is a substring of } w\}$. Give a finite automaton (you can use a DFA, an NFA, or an ε-NFA) that accepts L_1, and show informally why it is correct.

(b) (6 marks) Consider the language $L_2 = \{w \in \{0,1\}^* \mid 010 \text{ is not a suffix of } w\}$. Give a regular expression for L, and show informally why it is correct. **Hint**: there is a fairly simple answer to this question that avoids grief.

3. (12 marks): **A variant on NFAs**
A *thresholded DFA*, or TDFA, M, is a 6-tuple $M = (Q, \Sigma, \delta, q_0, F, \theta)$, with two changes from ordinary DFAs.

- A counter c_q is kept for each state q of the number of times that state has been exited during the FA's computation. These are initialized to 0 at the beginning of the execution of the FA on a word w, and when transitioning out of state q, its counter is incremented. For example, after reading the first input letter and following the appropriate transition from q_0, $c_{q_0} = 1$.
- θ is a vector of thresholds for each state in Q. If, for a given state q, we are about to take a transition from state q and c_q is already equal to θ_q, the TDFA instead terminates the computation and rejects the input word. All of the θ_q values are finite.

(a) (6 marks) Is it true that every regular language L is the language of an TDFA? Prove your answer.

(b) (6 marks) Prove that every TDFA accepts a regular language.