CPSC 313 Spring 2016

University of Calgary
Faculty of Science
Final Exam

2 July 2016

Time available: 180 minutes.

No books or calculators are permitted (two pages of notes are permitted).

Write answers in this booklet only.

Do not open the exam until you are told to do so.

Name: ______________________________

1) **Don't panic!** Partial credit is available.
2) Please give succinct, unambiguous, and well-phrased answers to the problems on the assignment. These qualities will be taken into account in the assessment.
3) There are 46 points in total and 7 bonus points. Full answer is 45 points.
4) Please read the entire exam before writing anything. Please ask for clarification if any question is unclear.
5) Please note that points may be deducted for unintelligible handwriting.
6) You may use the blank pages at the end of the exam if you need more space for your answers. Please indicate clearly when your answers are continued on these pages.
7) For your convenience, the last page contains a collection of information.
8) **Good luck!**

Problem	Possible score	Score
1	10	
2	16	
3	8	
4	6	
5	6	
6	4 bonus	
7	3 bonus	

Problem 1 True/False (10 points, 1 point each) Answer True or False. The languages are over the alphabet $\Sigma = \{0, 1\}$. No justification required.

	Question	True	False
1	The CFG with the rules $S \to SS \mid Sb \mid a \mid ab$ is ambiguous.		
2	If A is regular and $B \subseteq A$, then B is regular.		
3	If A reduces to B and B is undecidable, then A is undecidable.		
4	$(0+1)^* = 0^* + 1^*$.		
5	The regular expression $(0+11+10)^*1^*$ generates all binary strings		
6	If a language L is recognized by an NFA, then $\bar{L}$ is recognized by an NFA.		
7	If A is regular and B is context-free, then $A \cup B$ may not be context-free.		
8	If L is undecidable, then L is infinite.		
9	L is decidable if and only if its complement $\bar{L}$ is undecidable.		
10	If A is regular and $A \cup B$ is regular, then B is regular		

Problem 2 Short answer questions (16 points, 2 points each)

1) Is it true that, for all languages L_1 and L_2, we have $(L_1^* \cap L_2^*)^* = (L_1 \cap L_2)^*$? Give a short proof if you believe the statement is true or give a counter-example if you believe the statement is false.

2) Give a regular language over $\Sigma = \{a, b\}$ with a one-state NFA but no one-state DFA. Briefly justify your answer.

3) Give a regular expression for the language consisting of strings over $\{0, 1\}$ in which the number of 0s and the number of 1s are either both even or both odd.

4) Give a context-free grammar for the language L consisting of strings over $\{0, 1\}$ where the number of 0's is congruent with 2 modulo 5. For example $001 \in L, 1010 \in L$, but $0 \notin L, \epsilon \notin L$ and $10001 \notin L$. Explain the role of each non-terminal in your grammar.

5) We define over the alphabet $\Sigma = \{0, 1\}$ the language L consisting of all possible strings except the string 0110. Describe a DFA for L.

6) We define over the alphabet $\Sigma = \{0, 1\}$ the language L consisting of all possible strings except the string 0110. Write a regular expression for L.

7) Give a context-free grammar for the language $L = \{a^i b^j c^k \mid i + k < j\}$. Explain the roles of each non-terminal in your grammar.

8) Give an NFA for the language $L = \{0^n 1^m 2^p \mid p \equiv m + n \pmod 2\}$. For example $\epsilon \in L, 01 \in L, 22 \in L, 001112 \in L$, but $0 \notin L$ and $012 \notin L$.

Problem 3 Turing machine construction (8 points)

Describe a two-tape Turing machine that computes the function $\lceil \log_2 n \rceil$. Given on the first tape as input the string 1^n (for any positive integer n), your machine should output the string $1^{\lceil \log_2 n \rceil}$ on its second tape. For example, given the input string 1111111111111 (thirteen 1's), your machine should output the string 1111 because $2^3 < 13 \leq 2^4$. Test your machine on the inputs 1111 and 11111.

Your description should be at the level of the description in the class for the Turing machine that accepts the set $\{1^n \mid n \text{ is a power of } 2\}$. In particular, do not give a list of transitions and do not draw state diagrams.

(Blank page for continuing your answers to problem 3).

Problem 4 NFA design (6 points).
Let $L \subseteq \Sigma^*$ be an arbitrary regular language. Prove that the language

$$\textsc{stubs}(L) = \{x \mid xy \in L \text{ for some string } y \in \Sigma^*\}$$

is also regular. If you construct one machine $M' = (Q', \Sigma, \delta', q_0', F')$ from another machine $M = (Q, \Sigma, \delta, q_0, F)$, you must give precise definitions for Q, δ', q_0', and F'.

(Blank page for continuing your answers to problem 4).

Problem 5 Reduction (6 points). Prove that the language

$$\textsc{Include} = \{\langle M_1, M_2\rangle\} \mid M_1, M_2 \text{ are Turing machines and } L(M_1) \subseteq L(M_2)\}.$$

is undecidable.

(Blank page for continuing your answers to problem 5).

Problem 6 Enumeration and reduction (4 bonus points: 2 + 2). Consider the language

$$L = \{\langle M\rangle \mid M \text{ is a Turing machine such that, for all strings } x, M \text{ on input } x \text{ halts within } |x|^2 \text{ steps } \}.$$

Here $|x|$ denotes the length of a string x. We prove that L is not recognizable, but $\bar{L}$ is recognizable.

a) Design a recognizer for $\bar{L}$.

b) Prove that $\overline{\textsc{Halt}} \leq_m L$.

(Full bonus points to complete solutions. No points to partial solutions.)

(Blank page for continuing your answers to problem 6).

Problem 7 NFA design (3 bonus points).
This problem is intended to be a challenge.

Let $L \subseteq \Sigma^*$ be an arbitrary regular language. Prove that the language

$$\text{REFLECT}(L) = \{w \in \Sigma^* \mid ww^R \in L\}$$

is also regular. If you construct one machine $M' = (Q', \Sigma, \delta', q_0', F')$ from another machine $M = (Q, \Sigma, \delta, q_0, F)$, you must give precise definitions for Q, δ', q_0', and F'. (Full bonus points to complete solutions. No points to partial solutions.)

(Blank page for continuing your answers to any problem).

Theorem 1. *Let L_1 and L_2 be regular languages over the alphabet Σ. Then the following languages are regular.*

1. $\overline{L_1} = \{w \in \Sigma^* \mid w \notin L_1\}$,
2. $L_1 \cup L_2$,
3. $L_1 \cap L_2$,
4. L_1^*,
5. $L_1 L_2$,
6. $L_1^R = \{w \in \Sigma^* \mid w^R \in L_1\}$.

Theorem 2. *Any finite language is regular.*

Theorem 3. *Let $L_1 = \{w \in \{0,1\}^* \mid w = w^R\}$ and $L_2 = \{0^n 1^n \mid n \geq 0\}$. Then L_1 and L_2 are not regular.*

Theorem 4. *The language $\text{HALT} = \{\langle M, w\rangle \mid M \text{ is a Turing machine and } M \text{ halts on } w\}$ is undecidable.*

Theorem 5. *The language $\overline{\text{HALT}} = \{\langle M, w\rangle \mid M \text{ is a Turing machine and } M \text{ does not halt on } w\}$ is not recognizable.*

Example of a reduction

Theorem 6. *The language $\text{REV} = \{\langle M\rangle \mid M \text{ is a Turing machine and } M \text{ accepts } w^R \text{ whenever it accepts } w\}$ is undecidable.*

Proof. Assume we can decide the language REV. We show that this implies we can decide HALT. Given **any** input $\langle M, w\rangle$ to the HALT problem, we construct the machine M' that, on any input y works as follows.

1. if $y = 01$, accept.
2. if $y \neq 10$, reject.
3. if $y = 10$, simulate M on w and accept if M halts.

(We note that, when we construct M', we have access to M and w; in other words, M and w are hard-coded into the states of M'). If M halts on w, then $L(M') = \{01, 10\}$, so $\langle M'\rangle \in \text{REV}$. Conversely, if $\langle M, w\rangle \notin \text{HALT}$, then $L(M') = \{01\}$, so $\langle M'\rangle \notin \text{REV}$. This shows that the mapping $f(\langle M, w\rangle) = \langle M'\rangle$ is a reduction from HALT to REV. If we can decide if a machine belongs to REV, then we can apply this decision procedure to M', and this would allow us to tell whether M halts on w. Thus we would be able to decide HALT, contradiction. $\square$