CPSC 313 (Fall 2012) L01

University of Calgary

Faculty of Science

Final Examination

December 11, 2012.

Time available: 180 minutes.

No books or calculators are permitted (two 8.5×11 sheets of notes are permitted).

Write answers in this booklet only.

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

Name: ____

There are 6 (six) problems in total. There are $22 + 2$ points (22 points and 2 bonus points) in total. Full answer is 20 points.

For your convenience, the last page (page 16) contains a collection of information.

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.

Please note that points may be deducted for untidy handwriting.

Problem	Possible score	Score
1	6	
2	4	
3	5	
4	2	
5	5	
6	2 bonus	
Total	22 + 2 bonus	

1. Short Answers (6 points)

1. Give a regular expression for the language $L_1 = \{w \in \{a, b\}^\star \mid |w|_b \geq 2\}$.

2. Give a right-linear grammar for the regular language $L_2 = \{a^i b c^j \mid i, j \geq 0\}$.

3. Give a context-free grammar for the language $L_3 = \{a^i b^i c^j d^k \mid 0 \leq i \text{ and } 0 \leq j \leq k\}$.

4. Give a context-free grammar for the language $L_4 = \{a^i b^j c^k \mid i \leq j \leq i + k\}$.

5. Give a minimal DFA for the language $L_5 = \{a, aba, abc\}$.

6. Give a minimal DFA for the language $L_6 = \{a, aba, cba\}$.

2. One-way Turing Machines (4 points)

A one-way Turing Machine is a 7-tuple $M = (Q, \Sigma, \Gamma, \delta, q_0, \Box, F)$ defined identically to a standard Turing Machine except the tape head can move right and stay, but cannot move left.

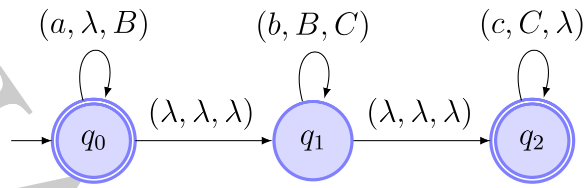
1. Give a language L_1 that **can** be decided be a one-way Turing Machine. Briefly outline a one-way Turing Machine that decides L_1.

2. Give a language L_2 that can **not** be decided by a one-way Turing Machine. Justify your answer.

3. Give a characterization of the class of languages that can be decided by one-way Turing Machines. Justify your answer.

3. Finite Automata equipped with a sack (5 points)

A Pushdown Automata (PDA) is a finite automata equipped with a stack. Now, define a Sack Finite Automata (SFA) as a finite automata equipped with a **sack**. An SFA operates similar to a PDA, except the transition function δ can remove any symbol in the sack. There is no order of the symbols in the sack. The production $\delta(q_i, a, B) = (q_j, C)$ means that from state q_i, we can read an 'a' of the input, take a symbol 'B' out of the sack, drop a 'C' into the sack, and go to state q_j. An SFA accepts an input string w if it ends in a **final accept state** with an **empty sack**.



1. Argue that the SFA above accepts the language $L_1 = \{a^i b^i c^i \mid i \geq 0\}$.

2. Give an SFA that accepts the language $L_2 = \{w \in \{a, b, c\}^* \mid |w|_a = |w|_b = |w|_c\}$. Argue that your SFA is correct.

3. Argue that if L_{reg} is regular and L_{SFA} is accepted by an SFA, then $L_{\text{reg}} \cap L_{\text{SFA}}$ is also accepted by an SFA.

4. Languages (2 points)
No justification required.

1. Give a language $L_1 \in \mathsf{CF} \setminus \mathsf{REG}$.

$L_1 =$ ______________________________

2. Give a language $L_2 \in \mathsf{REC} \setminus \mathsf{CF}$.

$L_2 =$ ______________________________

3. Give a language $L_3 \in \mathsf{RE} \setminus \mathsf{REC}$.

$L_3 =$ ______________________________

4. Give a language $L_4 \in \text{co-}\mathsf{RE} \setminus \mathsf{RE}$.

$L_4 =$ ______________________________

5. Language classification (5 points)

Answer **exactly one** of four possible answers CF, REC, RE, None to each of the following languages. Here CF means context-free, REC recursive, and RE recursive enumerable. Answer None if the language is not recursive enumerable. No justification required.

	Question	CF	REC	RE	None
1	$\{\langle M\rangle \mid$ TM M accepts exactly three strings$\}$				
2	$\{\langle M, w, i\rangle \mid M$ is TM and $w^i \notin L(M)\}$				
3	$\{a^i \mid$ there is some TM that loops forever on $a^i\}$				
4	$\{a,b\}^* \setminus \{a^i b^i a^i \mid i \geq 0\}$				
5	$\{ww^R w^R w \mid w \in \{a,b\}^*\}$				
6	$\{\langle i, j\rangle \mid L(M_i) \neq L(M_j)$ for the i^{th} and j^{th} TMs$\}$				
7	$\{a^i \mid i$ is not divisible by $7\}$				
8	$\{\langle w_1, w_2\rangle \mid$ either $w_1 = \lambda$ or $w_2 = \lambda\}$				
9	$\{\langle M\rangle \mid L(M) = \Sigma^\star$ and M is a TM$\}$				
10	$\{\}^*$				

6. Reduction (2 bonus points)
Prove that $L \leqslant_m \mathsf{Halt}$ where $L = \{0110\}$.

(This page is intentionally left blank)

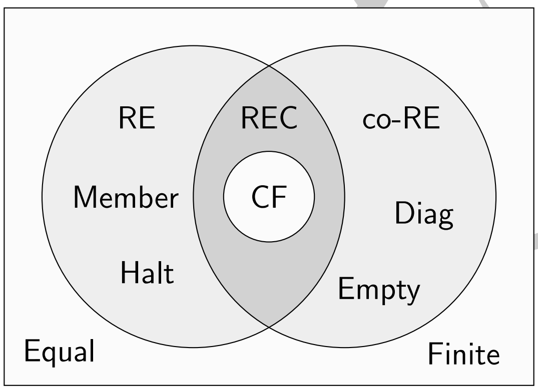
(This page is intentionally left blank)

(This page is intentionally left blank)

	$x \in L$	$x \notin L$	
$L \in \mathsf{REC}$	halts & accepts	halts & rejects	Recursive
$L \in \mathsf{RE}$	halts & accepts	—	Recursive enumerable
$L \in \mathsf{co\text{-}RE}$	—	halts & accepts	

$L \in \mathsf{RE} \Leftrightarrow \overline{L} \in \mathsf{co\text{-}RE}$
$L \in \mathsf{REC} \Leftrightarrow L \in \mathsf{RE}$ and $\overline{L} \in \mathsf{RE}$
$L \in \mathsf{RE} \Leftrightarrow \exists$ unrestricted grammar

L Recursive $=$ L Decidable $=$ χ_L Computable
L Not recursive $=$ L Undecidable $=$ χ_L Noncomputable



	REG	CF	REC	RE
$L_1 \cup L_2$	✓	✓	✓	✓
$L_1 \cap L_2$	✓	(1)	✓	✓
$\overline{L}$	✓	(2)	✓	(4)
$L_1 \setminus L_2$	✓	(3)	✓	(5)
$L_1 L_2$	✓	✓	✓	✓
L^*	✓	✓	✓	✓
L^R	✓	✓	✓	✓
prefix(L)	✓	✓	✓	✓
$h(L)$	✓	✓	(6)	✓
λ-free $h(L)$	✓	✓	✓	✓

$$\begin{array}{lcll}
\mathsf{Valid} & = & \{\langle M\rangle \mid M \text{ is a TM} & \} \\
\mathsf{Diag} & = & \{\langle M\rangle \mid M \notin L(M) & \& \ M \text{ is a TM} \} \\
\mathsf{Member} & = & \{\langle M, w\rangle \mid w \in L(M) & \& \ M \text{ is a TM} \} \\
\mathsf{Halt} & = & \{\langle M, w\rangle \mid M \text{ halts on input } w & \& \ M \text{ is a TM} \} \\
\mathsf{Empty} & = & \{\langle M\rangle \mid L(M) = \emptyset & \& \ M \text{ is a TM} \} \\
\mathsf{Finite} & = & \{\langle M\rangle \mid L(M) \text{ is finite} & \& \ M \text{ is a TM} \} \\
\mathsf{Equal} & = & \{\langle M_1, M_2\rangle \mid L(M_1) = L(M_2) & \& \ M_1 \text{ and } M_2 \text{ are TMs} \}
\end{array}$$

Language $A \subseteq \Sigma_A^*$ **reduces to** language $B \subseteq \Sigma_B^*$ if there is a computable function $f : \Sigma_A^* \to \Sigma_B^*$ such that $w \in A \Leftrightarrow f(w) \in B$ for all strings $w \in \Sigma_A^*$.

If A reduces to B, denoted $A \leqslant_m B$, then problem A is at most as difficult as problem B.
$(A \leqslant_m B$ and $A \notin \mathsf{RE}) \Rightarrow B \notin \mathsf{RE}$ (If problem A is hard, so is B.)
$(A \leqslant_m B$ and $B \in \mathsf{RE}) \Rightarrow A \in \mathsf{RE}$ (If problem B is easy, so is A.)
$(A \leqslant_m B) \Leftrightarrow (\overline{A} \leqslant_m \overline{B})$
Transitivity: $\big(A \leqslant_m B$ and $B \leqslant_m C\big) \Rightarrow A \leqslant_m C$