CPSC 313 (Winter 2016) L01

University of Calgary

Faculty of Science

Final Examination

April 18, 2016.

Time available: 180 minutes.

No books or calculators are permitted (two lettersized 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 7 (seven) problems in total. There are 33 points and 6 bonus points in total. Full answer is **33** points.

For your convenience, the last two pages (pp. 17–18) contain a collection of information.

You may use the blank pages in this 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	14	
2	5	
3	6	
4	3	
5	5 + 1 bonus	
6	3 bonus	
7	2 bonus	
Total		

1. Short answers (14 points)

1. Give a minimal DFA for the language $L(\boldsymbol{r})$ where $\boldsymbol{r} = \boldsymbol{a}(\boldsymbol{b}^*)\boldsymbol{a} + \boldsymbol{b}^*$.

2. Give a PDA for the language $L = \{a^i b^j c^k \mid j = i + k \text{ and } i, j, k \geq 0\}$.

3. Give an NFA for the *complement* of the language $L = \{abba\}$ over the alphabet $\Sigma = \{a, b\}$.

4. Give a regular expression that is as simple as possible for the language
$L = \{w \in \{a, b\}^* \mid w \text{ does not contain the substring } aa \text{ nor the substring } bb\}$.

5. Let M be a Turing Machine that halts on all inputs in at most 20 steps. Explain why $L(M)$ is necessarily **regular**.

6. Give a context-free grammar for the language $L = \{a^i b^j c^i d^k e^k \mid i, j, k \geq 0\}$.

7. The following theorem is true, but its "proof" is flawed. Precisely and succinctly identify a flaw in the proof and explain why it is a flaw. If there are more than one flaw, identify and explain them all.

Theorem 1 *The language* $L = \{a^i b^i c^i d^i \mid i \geq 0\}$ *is not context-free.*

Proof Let $L_1 = \{a^i b^j c^j d^i \mid i, j \geq 0\}$. Then L_1 is context-free. Let $L_2 = \{a^i b^j c^i d^j \mid i, j \geq 0\}$. Then L_2 is not context-free, and hence $L_1 \cap L_2$ is neither context-free. But $L_1 \cap L_2$ is equal to L, and hence L is not context-free. □

(This page is intentionally left blank)

2. A language not context-free (5 points)

Show that the language

$$L = \{a^i b^j c^k a^i b^j c^k \mid i \geq 0, j \geq 1, k = 1\}$$

is not context-free. You may, without proof, use the Pumping Lemma and the closure properties for the context-free languages.

(This page is intentionally left blank)

3. Reduction (6 points)

Let

$$\mathsf{Even} = \{\langle M\rangle \mid M \text{ is a TM and } L(M) \text{ contains only strings of even length}\}.$$

Show that $\mathsf{Halt} \leqslant_m \overline{\mathsf{Even}}$. That is, show that the halting problem reduces to the *complement* of the language Even.

(This page is intentionally left blank)

4. True or False (3 points)

Answer True or False. The languages are over the alphabet $\Sigma = \{0, 1\}$. No justification required.

	Question	True	False
1	If L is recursive, then so is the complement $\overline{L}$		
2	If $L \notin \mathsf{co\text{-}RE}$ then $L \in \mathsf{RE} \cup \mathsf{REC}$		
3	If L_1 is finite and $L_2 \in \mathsf{RE}$ then $L_1 \cap L_2 \in \mathsf{co\text{-}RE}$		
4	$L = \{\langle M \rangle \mid M \text{ is a TM and } L(M) = \emptyset\}$ is infinite		
5	If L is context-free then L is recursive		
6	$L = \{\langle M \rangle \mid M \text{ is DFA that accepts an infinite language}\}$ is recursive		
7	There is a recursive language L that is not regular		
8	$\mathsf{Halt} \leqslant_m \overline{\mathsf{Equal}}$		
9	$\mathsf{Empty} \not\leqslant_m \mathsf{Valid}$		
10	$\mathsf{Member} \leqslant_m \mathsf{Member}$		

5. Language classification (5 points + 1 bonus point)

Answer **exactly one** of four possible answers REC, RE, co-RE, None to each of the following languages. Answer REC if the language is recursive, answer RE if the language is recursive enumerable but not recursive, answer co-RE if the language is co-recursive enumerable but not recursive, and answer None if the language is none of the first three possible answers. No justification required. Below, M, M_1 and M_2 denote Turing Machines. The languages are over the alphabet $\Sigma = \{0, 1\}$.

	Question	REC	RE	co-RE	None
1	$\left\{\langle M, w\rangle \mid M \text{ is a TM that either loops on } w \text{ or accepts } w\right\}$				
2	$\left\{\langle M\rangle \mid M \text{ is a TM that has an odd number of final states}\right\}$				
3	$\left\{\langle M, w\rangle \mid L(M) = \{w\}\right\}$				
4	$\left\{w \mid w = \langle M\rangle \text{ for some TM } M \text{ and } w \in L(M)\right\}$				
5	$\left\{\langle M\rangle \mid \begin{array}{l} M \text{ is a TM that halts in a non-final} \\ \text{state on some input} \end{array}\right\}$				
6	$\left\{\langle M\rangle \mid L(M) = L(\mathbf{1}^*)\right\}$				
7	$\left\{w \mid w \text{ is rejected by some TM } M\right\}$				
8	$\left\{\langle M_1, M_2\rangle \mid L(M_1) \cap L(M_2) \neq \emptyset\right\}$				
9	$\left\{\langle M, w_1, w_2\rangle \mid \begin{array}{l} M \text{ is a TM that halts on } w_1 \text{ and} \\ \text{accepts } w_2 \end{array}\right\}$				
10	$L = \{10110\}$				
11	$\overline{\mathsf{Empty}} \cup \mathsf{Finite}$				
12	$\left\{\langle G\rangle \mid G \text{ is a context-free grammar and } L(G) \neq \emptyset\right\}$				

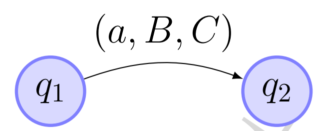
6. Reversal of context-free languages (3 bonus points)

The context-free languages are closed under reversal.

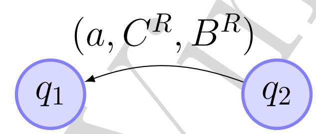
Theorem 2 *If L is context-free, then the reverse language L^R is also context-free.*

It is possible to prove this theorem using context-free grammars, by replacing each rule $A \to w$ with its reversal $A \to w^R$, where w^R denote string w reversed. Consider we instead want to prove the above theorem using PDAs. Here is our first flawed proof idea.

Proof [Flawed idea] Let $M = (Q, \Sigma, \Gamma, \delta, q_0, z, F)$ be a PDA such that $L(M) = L$. We construct a new PDA $M' = (Q, \Sigma, \Gamma, \delta', q_0, z, F)$ that is the same as M, except for the transition function δ'. Whenever we have a rule in M of the form $\delta(q_1, a, B) = (q_2, C)$ in δ, we include the rule $\delta(q_2, a, C^R) = (q_1, B^R)$ in δ'. In short, we replace each edge



with its reversed edge



□

This proof idea almost work, but not quite. We need to make some additional adjustments to the proof. Explain what these additional adjustments are. Make your arguments succinct. (Full bonus points to complete solutions. No points to partial solutions.)

(This page is intentionally left blank)

7. Closure property (2 bonus points)

Prove that if $L_1 \in \mathsf{REC}$ and $L_2 \in \mathsf{RE}$, then $L_1 \setminus L_2 \in \mathsf{co\text{-}RE}$. Make your proof succinct. (Full bonus points to complete solutions. No points to partial solutions.)

Theorem 3 (Pumping Lemma for Context-Free Languages) *Let L be a context-free language. Then there*

$$\textit{exists integer } m \in \mathbb{Z} \textit{ such that for all } w \in L \text{ with } |w| \geq m,$$

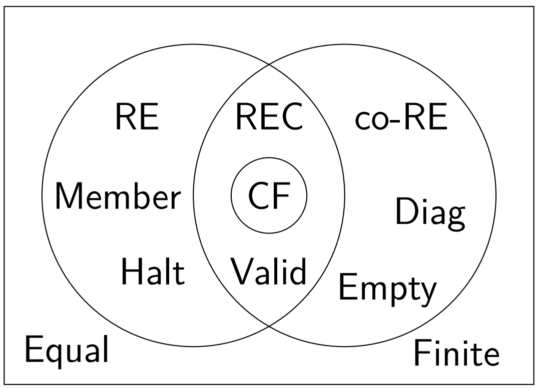
there exist strings u, v, x, y, and z, such that $w = u \cdot v \cdot x \cdot y \cdot z$ and

1. $|v\,x\,y| \leq m$,
2. $|v\,y| \geq 1$, *and*
3. *for all* $i \geq 0$, $u \cdot v^i \cdot x \cdot y^i \cdot z \in L$.

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

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

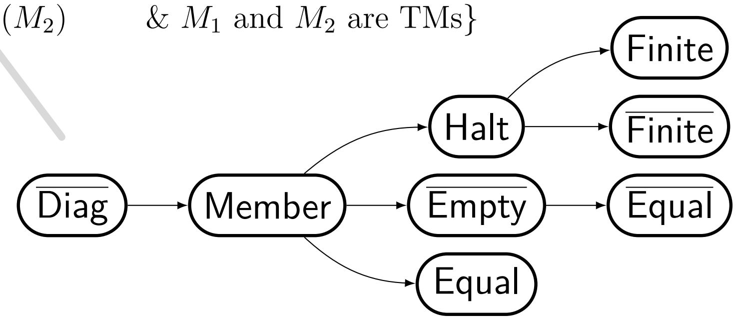
L Recursive $=$ L Decidable $=$ χ_L Computable
L Not recursive $=$ L Undecidable $=$ χ_L Noncomputable



	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 $h(L)$	✓	✓	✓	✓

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

$\overline{\textsf{Diag}} \le_m \textsf{Member}$
$\textsf{Member} \le_m \overline{\textsf{Empty}}$
$\textsf{Halt} \le_m \overline{\textsf{Finite}}$
$\textsf{Halt} \le_m \textsf{Finite}$
$\textsf{Member} \le_m \textsf{Equal}$
$\textsf{Member} \le_m \textsf{Halt}$
$\overline{\textsf{Empty}} \le_m \overline{\textsf{Equal}}$



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

******** PH/PH ********