NAME: ______________________________

SPIRE ID: ______________________________

COMPSCI 501
Formal Language Theory
Midterm Spring 2024

D. A. M. Barrington 3 April 2024

DIRECTIONS:

- Answer the problems on the exam pages.
- There are seven problems on pages 2-7, some with multiple parts, for 100 total points plus 5 extra credit. Final scale will be determined after the exam.
- The supplemental page 8 has definitions for your use and should not be handed in.
- If you need extra space use the back of a page – both sides will be scanned.
- No books, notes, calculators, or collaboration.

1	/20
2	/30
3	/10
4	/10
5	/10
6	/10
7	/10+5
Total	/100+5

Question 1 (20): These are ten true/false statements, with no justification needed or wanted (2 points each):

- (a, 2) The intersection of any finite collection of CFL's is decidable, but it is not necessarily a CFL itself.

- (b, 2) Let X and Y be two languages such that there exists some function f such that for all strings w, $(w \in X) \leftrightarrow (f(w) \in Y)$. Then it must be the case that $X \leq_m Y$.

- (c, 2) There exists an undecidable language with a unary alphabet, that is, with $|\Sigma| = 1$ where Σ is the input alphabet.

- (d, 2) Let A and B be two languages such that B is undecidable, and there exists a reduction showing $A \leq_m B$. Then A must also be undecidable.

- (e, 2) Recall that an n-bit string w is **incompressible** if its Kolmogorov complexity satisfies $K(w) \geq n$. If v and w are both incompressible n-bit strings, then the string vw is also incompressible.

- (f, 2) Let Σ be any non-empty alphabet. Then the language $L = \{a^n b^n : n \geq 0, a \in \Sigma, b \in \Sigma\}$ must not be regular.

- (g, 2) It is possible to convert any context-free grammar G into an NFA N such that $L(G) = L(N)$.

- (h, 2) Let f be any function from $\mathbf{N} \times \mathbf{N}$ to $\{0, 1\}$, where $\mathbf{N}$ is the set of natural numbers. Then there does not exist a function d from $\mathbf{N}$ to $\{0, 1\}$ such that for any n, $d(n) \neq f(n, n)$.

- (i, 2) For any string w, define a TM to be a w-**printer** if, on any input, it halts with w on its tape. Then there exists some Turing machine M such that, on any input w, M halts with the description of some w-printer on its tape.

- (j, 2) Let M be any Turing machine and let t be any partial function computed by it, so that for any input x, $t(x)$ is the string left on M's tape if it halts (if it doesn't halt, then $t(x)$ is not defined). Then there exists a Turing machine R such that for any input string w, R accepts w if and only if the value $t(w)$ is not defined.

Question 2 (30): These are five true-false questions, with brief justification required. Three points for each correct boolean answer, and up to three points per question for the justification:

- (a, 6) Let c be any positive natural. A c-PDA is a (nondeterministic) pushdown automaton such that its stack never contains more than c characters. Then the language of any c-PDA is regular.

- (b, 6) Let X be any non-empty TD language whose complement $\overline{X}$ is also non-empty. Then $\overline{X} \leq_m X$.

- (c, 6) Let X be any non-empty TR language whose complement $\overline{X}$ is also non-empty. Then $\overline{X} \leq_m X$.

- (d, 6) Recall that PCP is the language of finite sets of dominoes that contain a match. Then the language PCP is TR-complete.

- (e, 6) The language EQ_{CFG} is co-TR, but not TD.

Definitions: Some of Questions 3-5 deal with two new definitions. If X is any language over some alphabet Σ, we define the **cube root** of X, called $CR(X)$, to be the language $\{w : www \in X\}$. Similarly, the **cube** of X, called $Cube(X)$, is the language $\{www : w \in X\}$.

Question 3 (10): Prove that if X is any regular language over any finite alphabet Σ, then $CR(X)$ is also a regular language.

Question 4 (10): Prove that there exists a context-free language Y (over the alphabet $\{0, 1\}$) such that $Cube(Y)$ is not context-free.

Question 5 (10): Here are two more problems about the cube root and cube languages defined above:

- (a) Prove that if Z is a TR language, prove that both $CR(Z)$ and $Cube(Z)$ are also TR languages.

- (b) Prove that if Q is the language of an LBA, then $CR(Q)$ is also the language of an LBA.

Question 6 (10): In this problem we define a **Silly TM** to be a one-tape machine with two read heads and no ability to change letters on its tape. On a given time step, each head may either move right or stay where it is. Prove that the language E_{SillyTM} is undecidable.

Question 7 (10+5): Given a collection of Turing machines $\{M_i\} = \{M_0, M_1, M_2, \ldots\}$, $\{M_i\}$ is defined to be a **computable collection** if there exists a computable function $q : \{1\}^* \to \Sigma^*$ such that $q(1^n) = \langle M_n \rangle$ (that is, q takes a natural number and maps it to the description of the n'th Turing machine in the collection). In this case we also call $\{L(M_i)\}$ a **computable collection of TR languages**.

- (a) Is the union of any countable collection of TR languages necessarily TR? Prove your answer.

- (b) Is the union of any countable *computable* collection of TR languages necessarily TR? Prove your answer.

- (c, extra credit) Is the intersection of a countable computable collection of TR languages necessarily TR? Prove your answer.

Supplemental Page for COMPSCI Midterm Spring 2024

Language A_{TM} (similarly A_{REG}, etc.) Set of pairs (M, w) such that M accepts w

Language ALL_{TM}**:** (similarly ALL_{REG}, etc.) Set of machines that accept all possible strings over their alphabet

Computable Function: Function f from strings to strings such that some Turing machine, on any input w, always halts with $f(w)$ on its tape

Countable Set: Any set that is either finite or has a bijection with the set of natural numbers

Context-Free Grammar (CFG): Grammar where rules allow single non-terminals to be replaced by strings

Context-Free Language (CFL): Definable by a context-free grammar or a PDA

co-TR: A language A is co-TR if and only if its complement $\overline{A}$ is TR.

Language E_{TM}**:** (similarly E_{REG}, etc.) Set of machines with empty languages

Language EQ_{TM}**:** (similarly EQ_{REG}, etc.) Set of pairs of machines with equal languages, *i.e.*, $\{(M_1, M_2) : L(M_1) = L(M_2)\}$.

Linear Bounded Automaton (LBA): A machine like a one-tape Turing machine, but with no additional space to the right of its input.

Mapping Reduction ($\leq_m$**):** $A \leq_m B$ means that there exists a computable function f such that $\forall w : w \in A \leftrightarrow f(w) \in B$

Pushdown Automaton (PDA): Nondeterministic finite-state machine with an added stack

Language REG_{TM}**:** (similarly REG_{REG}, etc.) Set of machines M such that $L(M)$ is a regular language

Regular Language: Can be defined by a DFA, NFA, or regular expression

Turing Decidable (TD): Is the language of a TM that always halts

Turing Recognizable (TR): Is the language of any TM