CSE396, Fall 2017 **Final Exam** **May 16, 2017**

One notes binder allowed, no Internet, closed neighbors, 170 minutes. The exam has **seven** problems and totals 267 pts., subdivided as shown. Problem 6 includes a *choice*—you must do ANY TWO of its parts (a), (b), or (c). *Show your work*—this may help for partial credit. *Please write on these sheets only*—if you need more paper you may ask.

Notation and terminology are as in the text, except for the following cases in which lectures gave alternative terminology:

- *Turing-recognizable* language and *c.e. or r.e.* language are synonyms. The class of such languages is denoted by RE.
- *Decidable* language and *recursive* language are synonyms. The class of such languages is denoted by REC.
- Angle brackets $\langle M \rangle$ denote the encoding of a Turing machine M as a string over the alphabet ASCII. Fine details of such encodings are not important, and *languages using such encodings may be assumed not to be CFLs.* Similar remarks apply to the use of (e.g.) $\langle P, w \rangle$ to encode a program P and an input w as a single string.
- The notation $P(x)$ means "the computation of P on input x."

As usual, x^R stands for the reversal of the string x, and $\#c(x)$ stands for the number of times the character c occurs in x. For any character c and number n, c^n stands for a run of n consecutive c's. The complement of a language A is denoted by $\tilde{A}$, and ϕ stands for a Boolean formula.

You may cite theorems and facts that were covered in lectures and/or text without further proof, so long as the cited item is clearly stated and your use of it is clear. You may refer to the following (not exhaustive!) list of languages and their classifications:

$\{\, a^n b^n : n \geq 0 \,\}$	DCFL, but not regular
$\{\, x \in \{\, a, b \,\}^* : x = x^R \,\}$	CFL and co-CFL, but not a DCFL
$\{\, a^n b^n c^n : n \geq 0 \,\}$	Co-CFL and in P, but not a CFL
$\{\, a^m b^n a^m b^n : m, n \geq 0 \,\}$	Co-CFL and in P, but not a CFL
$\{\, ww : w \in \{\, a, b \,\}^* \,\}$	Co-CFL and in P, but not a CFL
$\{\, \langle G, w \rangle : G$ is a CFG and $w \in L(G) \,\}$	A_{CFG}: In P but not a CFL (or co-CFL).
$\{\, x\#y\#z : x = y \,\wedge\, y \neq z \,\}$	In P, but not CFL or co-CFL
$\{\, \phi : (\exists a \in \{\, 0, 1 \,\}^n)\ \phi(a) = 1 \,\}$	SAT: In NP, believed not in P
$\{\, \phi : (\forall a \in \{\, 0, 1 \,\}^n)\ \phi(a) = 1 \,\}$	TAUT: In $\mathsf{co\text{-}NP}$, not in NP unless $\mathsf{NP} = \mathsf{co\text{-}NP}$
$\{\,$ poly-time NTMs $N : \langle N \rangle \notin L(N) \,\}$	D_{NP}: Decidable but not in NP
$\{\, \langle M, w \rangle : M$ is a TM and accepts $w \,\}$	R.e. but not co-r.e.
$\{\, \langle M \rangle : M$ is a TM and does not accept $\langle M \rangle \,\}$	Co-r.e. but not r.e.
$\{\, \langle M \rangle :$ for all inputs x, $M(x)$ halts $\,\}$	Neither r.e. nor co-r.e.

(1) (45 pts.)

Classify each of the following languages $L_1, \ldots, L_9$ according to whether it is

(a) regular;

(b) a DCFL but not regular;

(c) a CFL but not a DCFL;

(d) a co-CFL but not a CFL;

(e) in P but neither a CFL not a co-CFL;

(f) decidable but not in P;

(g) r.e. but not decidable; or

(h) not r.e.

You need not justify your answers, but *brief* justifications may help for partial credit. In the first few languages, $i, j, k, \ell \geq 1$.

1. $L_1 = \{\, a^i b^j c^k d^\ell : i \neq k \vee j \neq \ell \,\}$.
2. $L_2 = \{\, a^i b^j c^k d^\ell : i \neq j \vee k \neq \ell \,\}$.
3. $L_3 = \{\, a^i b^j c^k d^\ell : i \neq \ell \vee j \neq k \,\}$.
4. L_4 = any of the above but with '$\neq$' replaced by the relation "not congruent modulo 4."
5. $L_5 = \{\, 0^m 1^n 0^m 1^n : m, n \geq 1 \,\}$.
6. $L_6 = \{\langle M \rangle : M$ is a DTM with k states that does not accept $\langle M \rangle$ within $|\langle M \rangle|^k$ steps$\}$.
7. $L_7 = \{\langle P \rangle : P$ is a Java program and for all inputs x, $P(x)$ throws an `ArrayIndexOutOfBoundsException` $\}$.
8. $L_8 = \{\langle P \rangle : P$ is a Java program and there exists an input x such that $P(x)$ throws an `ArrayIndexOutOfBoundsException` $\}$.
9. $L_9 = \{\langle G \rangle : G$ is a context-free grammar and $\epsilon \in L(G)\}$.

Please write your answers in this form: if L_{10} were the language of the Halting Problem, you could write "10. g" or "10. (g)" or (safest) "L_{10} is (g) r.e. but undecidable." *No justifications are needed.*

(2) 9 × 5 = 45 pts. *Multiple Choice*: Indicate clearly the one best answer for each in the form (number). (letter).

1. If A is a regular language with a PD set of size q, no bigger, and B is a regular language with a PD set of size r, no bigger, and $C = A \triangle B$, then:
   (a) C has a PD set of size $q + r$;
   (b) All PD sets for C have size at most $q + r$;
   (c) All PD sets for C have size at most qr;
   (d) C can have an infinite PD set.

2. If two different variables in a context-free grammar G each derive ϵ, then:
   (a) G must be ambiguous;
   (b) $\epsilon \in L(G)$;
   (c) Converting G to G' in Chomsky normal form makes $L(G') \neq L(G)$;
   (d) None of the above necessarily happens.

3. To prove a language E is co-c.e. but not decidable, we could first give a TM M such that $L(M) = \tilde{E}$. Then we would (and would be able to):
   (a) Show that $L(M) = \tilde{E}$ leads to a contradiction;
   (b) Give a mapping reduction from D_{TM} to E;
   (c) Give a mapping reduction from A_{TM} to E;
   (d) Reduce E to the Halting Problem.

4. The concatenation $\emptyset \cdot \Sigma^*$ equals:
   (a) $\emptyset$;
   (b) $\{\epsilon\}$;
   (c) Σ^*;
   (d) None of the above.

5. The intersection of a regular language and a nonregular language can be:
   (a) Regular;
   (b) Nonregular;
   (c) Undecidable;
   (d) Any of the above.

6. If A is a DCFL and r is a regular expression, then $L(N) \cap L(r)$ is:
   (a) Always Regular;
   (b) Always a DCFL;
   (c) Always a CFL that is not a DCFL; or
   (d) Possibly not a CFL.

7. If $A \subseteq \{0,1\}^*$ is nonempty, then A^* always:
   (a) Includes some word of the form ww with $w \neq \epsilon$;
   (b) Includes some word of the form $wwwwwwwwwwwwwwwwwwww$ with $w \neq \epsilon$;
   (c) Includes ϵ.
   (d) All of the above.

8. In the CFG $G = S \longrightarrow aS \mid Sa \mid b$,
   (a) The string aba is ambiguous;
   (b) The string aab is ambiguous;
   (c) The string bb is ambiguous;
   (d) No string in $L(G)$ is ambiguous.

9. A DFA M such that $L(M) = \{x \in \{a,b\}^* : \#a(x) \text{ is even or } \#b(x) \text{ is odd}\}$ needs how many states?
   (a) Four, including a dead state;
   (b) Four, without a dead state;
   (c) Five, without a dead state;
   (d) Infinity: the language is nonregular.

(3) 5 × 3 = 15 pts. *True/false*. You must write out the word `true` *or* `false` *in full*, and justifications are not needed.

(a) *True/false?:* If A is a CFL and B is a DCFL, then $A \cap B$ is always a CFL.

(b) *True/false?:* A is a CFL and B is a DCFL, then $A \cdot B$ is always a CFL.

(c) *True/false?:* If A and B are Turing-recognizable, then so is $A \cap B$.

(d) *True/false?:* If A and B are Turing-recognizable, then so is their symmetric difference $A \triangle B$.

(e) *True/false?:* It is not known whether SAT polynomial-time many-one reduces to TAUT.

(4) (78 pts. total)

Take $\Sigma = \{ s, d, 0 \}$ where we interpret s as a "spear," d as a "dragon," and 0 as an "empty chamber." We view strings $x \in \Sigma^*$ as linear "dungeons" and play the spears-and-dragons game according to these new rules:

- The player P enters the dungeon with 0 spears.
- If P holds k spears and enters a room with a spear, then P now holds $\min\{ 2, k+1 \}$ spears. That is, P may never hold more than 2 spears.
- If P enters a room with a dragon and is holding 2 spears, P kills it but both spears are left in the dragon's body, so P enters the next room with 0 spears.
- If P enters a room with a dragon and is holding 1 spear, P can momentarily "stun" the dragon but survives only if the current room is the end of the dungeon or the next room holds a spear. In the latter case, P uses that spear to finish off the dragon but is again left with 0 spears entering the room after that.
- Otherwise, that is if P sees a dragon when holding 0 spears or with 1 spear and the next room is empty or has another dragon, P dies.

Let L be the language of $x \in \Sigma^*$ such that P exits x while still alive—it doesn't matter how many spears P has at the end. Here are some strings in L: ϵ, ssd, sds, sd, $ssdsd$ (in the last two, P is "saved by the exit"). Here are some strings not in L: d, $sd0$, $ssssdd$, $sdssd0$. In the third, P can hold only 2 spears before facing the dragons, so has none with which to fight the second one. In the last one, P had to use the second spear to kill the first dragon, and so had only one spear in hand to face the second dragon, and the emptiness of the following room doomed P before reaching the exit.

(a) For each of the following strings, say yes/no whether it belongs to L (2 pts. each):

(i) $sds0$ (ii) $sd0sss$ (iii) $sssdsd$ (iv) $sssdsdsd$ (v) $s0ssdsd0$ (vi) dx, for all $x \in \Sigma^*$.

(b) Design a DFA M such that $L(M) = L$. For full credit, you must give a well-commented arc-node diagram, including saying how many spears P has in any state. (18 pts.)

(c) Find a regular expression r such that $L(r)$ equals **the complement of** L. You should work by complementing your DFA M and applying the algorithm from class—you may take some shortcuts on the latter. (18 pts.)

(d) Now define L' by changing the rules to allow P to hold *any number of spears*. Now strings like $ssssdd$ and $sssdds$ belong to L', but $sssdd0$ still does not. Take $G = (V, \Sigma, R, I)$ to be the context-free grammar with variables $V = \{ I, B, E \}$ and rules $R =$

$$\begin{aligned} I &\longrightarrow sBdI \mid sEdsB \mid EsI \mid E \\ B &\longrightarrow sI \mid BdIs \\ E &\longrightarrow 0E \mid \epsilon. \end{aligned}$$

Attempt to prove that $L(G) \subseteq L'$. BUG! It's *false*. However, it is possible to change **one occurrence of** a variable in **one** rule into **another variable**, making a grammar G' such that $L(G') \subseteq L'$ becomes true. Find the change and prove $L(G') \subseteq L'$. (27 pts. total)

(e) If we add the rule $B \to BI$, does B then derive ϵ? (3 pts.)

(5) (24 pts.)

Consider the following decision problem:

PURE A'S
Instance: A context-free grammar $G = (V, \Sigma, R, S)$ with $\Sigma = \{a, b\}$.
Question: Is $L(G) \subseteq a^*$?

Give a decision procedure for this problem. For full credit, give enough detail to show that it belongs to the class P (polynomial time).

(6) (36 pts.)

Consider the following threee languages over the alphabet $\Sigma = \{a, b\}$:

$$\begin{aligned} L_1 &= \{\, x : |x| \text{ is odd and the middle char is a `}b\text{'} \,\}; \\ L_2 &= \{\, x : x = x^R \,\}; \\ L_3 &= \{\, x : \#a(x) = \#b(x) \,\}. \end{aligned}$$

On Prelim II, you showed that $L_2 \cap L_3$ is not a CFL. Which of $L_1 \cap L_2$ and $L_1 \cap L_3$ is a CFL? Do ANY TWO of the following—each is 18 pts.:

(a) For the one that is a CFL, design a context-free grammar for it.

(b) For the one that is not a CFL, prove it via the CFL Pumping Lemma; or

(c) Prove that L_1 is not regular using the Myhill-Nerode Theorem.

(7) (18+6 = 24 pts.)

Show that the following decision problem is undecidable, either by a mapping reduction or by showing that a problem we've already proved undecidable is a special case of it.

INSTANCE: A Turing machine M and a regular expression r.

QUESTION: Does M accept some string that does not match r?

Also answer: Is the language of this problem Turng recognizable? Justify your answer briefly.

END OF EXAM.