Name: ____

1. (1 pt.)
   - ○ **Read all material carefully.**
   - ○ *If in doubt whether something is allowed, ask, don't assume.*
   - ○ You may refer to **your** books, papers, and notes during this test. (No sharing of material.)
   - ○ **E-books** may be used **subject to the restrictions** noted in class. (Briefly, do only those things with an e-book that are just as easily done with a physical book.)
   - ○ **Computers of any kind** (including tablets, phones, and similar devices) are **not permitted** except when used exclusively as e-book readers.
   - ○ **Network access** of any kind (cell, voice, text, data, ...) is **not permitted**.
   - ○ Write and draw neatly and carefully. Ambiguous or cryptic answers receive no credit.
   - ○ Use class and textbook **conventions** for notation, algorithmic options, etc.
   - ○ Questions that ask for **explanations, proofs, etc.** allocate a sizable fraction of points to those. (Answers missing those will score very poorly.)
   - ○ Budget your **time**, noting that *number of points = number of minutes.*

Write your name in the space provided above.
Do not write anything else on this page.

WAIT UNTIL INSTRUCTED TO CONTINUE TO REMAINING QUESTIONS.

(Do not view any other pages.)

Do not write on this page.
(It is for use in grading only.)

Q	Full	Score
1	1	
2	9	
3	10	
4	15	
5	15	
total	50	

2. (9 pts.) Recall this definition:

   > Given languages L_1 and L_2, define $L_1L_2 = \{xy \mid x \in L_1 \wedge y \in L_2\}$.

   Prove or disprove: If L_1L_2 is a prefix-free language then L_1 and L_2 are also prefix-free. [Hint: Examine boundary cases carefully.]

3. (10 pts.) Is the following a regular expression (*RegExp*)?

$$(((a+b)^*) \circ (c+(b^*)))$$

If so then briefly outline its derivation from the formal definition of RegExps (for example, by drawing a parse tree). Otherwise, (1) prove as precisely as possible that it is not a RegExp and (2) modify it as little as possible to generate a valid RegExp for use in later questions.

4. (15 pts.) Provide a (possibly nondeterministic) finite-state automaton that is equivalent to the regular expression of Question 3. You are *not* required to use the mechanical method of conversion from regular expressions to NFAs, although you may. Present your automaton clearly using the usual diagrammatic format. (There is no need for a formal definition as a 5-tuple.) Prove correctness as precisely as possible. See also Question 5.

5. (15 pts.) Repeat Question 4 using a deterministic (instead of nondeterministic) finite-state automaton. You are *not* required to use the mechanical method of conversion from NFAs to DFAs, although you may. Prove correctness as precisely as possible.

[additional space for earlier material]

[additional space for earlier material]