Midterm — Section 2

Instructions: Please answer the questions succinctly and thoughtfully. Good luck.
— Phil Rogaway

Name:

Signature:

On problem	you got	out of
1		45
2		30
3		25
Σ		100

1 Short Answer [45 points]

Let M_1 be an n_1-state NFA and let M_2 be an n_2-state NFA. Using the procedures given in class and in your text, how many states will be in the DFA M the language of which is $\overline{L(M_1) \cup L(M_2)}$? Explain your reasoning.

---

(2) An n-**operation regular expression** is a regular expression that uses a total of n operations—union, concatenation, or star. For example, $001^* \cup 1$ is 4-operation regular expression. Let $f(n)$ be the maximum number of states that you get in your NFA when, using the procedure given in class, you convert an n-operation regular expression into an NFA. Give a formula for $f(n)$

---

(3)

Complete the following, mathematically precise, definition, according to the conventions of our text

a **CFG** is a 4-tuple $G = ($ $)$ where:

(4) Recall that, for $L \subseteq \{0,1\}^*$, $\mathrm{PAL}(L) = \{x \in \{0,1\}^* : xx^R \in L\}$. Write a regular expression for $\mathrm{PAL}(0^*10^*)$.

---

(5) Describe a decision procedure (algorithm) to decide the following language: $\{\langle M \rangle :$ M is a DFA that accepts infinitely many strings of even length$\}$. Assume an alphabet of $\{0,1\}$.

---

(6) Write a CFG for the language $L = \{a^i b^j c^k :\ i \neq j \text{ or } j \neq k\}$.

(7) Use the pumping lemma for regular languages to prove that the following language is *not* regular: $L = \{w \in \{a, b\}\ w$ is an odd-length palindrome$\}$ is **not** regular.

---

(8) Is the following CFG ambiguous? Explain.

$S \rightarrow 0S \mid 1A$

$A \rightarrow 1S \mid 0A \mid \varepsilon$

---

(9) Carefully **state** the pumping lemma for context-free languages.

2 Justified True or False [30 points]

Put an **X** through the **correct** box. Where it says "Explain" provide a **brief** (but convincing) justification. No credit will be given to correct answers that lack a proper justification. Where appropriate, **make your justification a counter-example**. Throughout, we use L to denote a language (maybe regular, maybe not).

---

1. If L is regular then L is context free. **True** **False**

   Explain:

2. If $M = (Q, \Sigma, \delta, q_0, F)$ is an NFA and $F = Q$ then $L(M) = \Sigma^*$.

   Explain: **True** **False**

3. If $M = (Q, \Sigma, \delta, q_0, F)$ is an NFA and $M' = (Q, \Sigma, \delta, q_0, F')$, where $F' = Q - F$, then $L(M') = \overline{L(M)}$. **True** **False**

   Explain:

4. If L is context free then $\overline{L}$ is context free. **True** **False**

   Explain:

5. There is an algorithm to decide if a CFG G generates the string *abbaa*.

Explain: **True** **False**

6. If L is accepted by an NPDA then L is accepted by a 3-state NPDA. (Assume the conventions on PDAs adopted in lecture.) **True** **False**

Explain:

3 A Little Algorithm [25 points]

A **black box** for a language L is a device (a subroutine) B that, when called on a string x, answers **Yes** if $x \in L$ and **No** if $x \notin L$. Suppose I give you a black box B for some language $L \subseteq \{0,1\}^*$. I don't tell you what is L, but I do tell you that $L = L(M)$ for some DFA M having 50 or fewer states.

Describe an algorithm A to determine if $L(M)$ is **finite** or **infinite**. Explain why your algorithm works. You will be calling B as a subroutine. You may make any number of calls.