Harvard University Extension School
Computer Science E-121
Problem Set 3

Due Friday, October 9, 2015 at 11:59 PM.
Submit your solutions electronically on the course website, located at `https://canvas.harvard.edu/courses/4896/assignments`. On the site, use the assignments tab to find the correct problem set, then with the "submit assignment" button, upload the PDF file of your solution.

LATE PROBLEM SETS WILL NOT BE ACCEPTED.

See the syllabus for the collaboration policy.

Important Note: You must redownload cscie121.cls from Canvas for this problem set, or your .tex will not compile.

PROBLEM 1 (3+3+3+3+3+3 points, suggested length of 3/4 page)

For each of the following languages, state whether the language is regular or non-regular. If regular, state a regular expression that denotes the language. If non-regular, write a short proof to justify.

(A) $\{xyx^R : x, y \in \Sigma^* \}$
(B) $\{a^i b^j a^j b^i$: $i, j \geq 0\}$.
(C) $\{w : w$ is, for some $n \geq 1$, the decimal notation for $10^n\}$.
(D) $\{a^{2n} b^n : n \geq 0\}$
(E) $\{a^n b^n : n \geq 0$ and $n \neq 3i + 5j$ for any $i \in \mathbb{N}, j \in \mathbb{N}\}$ Hint: Try explicitly working out all the strings in the language.
(F) $\{R : R$ is a regular expression for a language over $\Sigma\}$ for some alphabet Σ.

PROBLEM 2 (2+4 points, suggested length of 1/2 page)

(A) A context-free grammar G is ambiguous if there exists a string $w \in L(G)$ with two distinct leftmost derivations in G. Show that the context-free grammar $G = (V, \Sigma, R, S)$, where $V = \{S, A, a, b\}$, $\Sigma = \{a, b\}$, and $R = \{S \to AA, A \to AAA, A \to bA, A \to Ab, A \to a\}$ is ambiguous, because aba has two different leftmost derivations in G.

(B) Prove that $L(G)$, where G is the context-free grammar in Part (A), is regular.

PROBLEM 3 (3+3+2 points, suggested length of 3/4 page)

$$L_1 = \{a^n b^m : m, n \geq 0, m \neq n\}$$
$$L_2 = \{a^n b^m a^m b^n : m, n \geq 0\}$$

(A) Construct a context-free grammar G_1 for L_1
(B) Construct a context-free grammar G_2 for L_2.
(C) Construct a context-free grammar for $L_1 L_2$.

PROBLEM 4 ($3 + 3 + (2)$ points, suggested length of $\frac{1}{2}$ page)

An *arithmetic progression* is a set of the form $\{p + qn : n \in \mathbb{N}\}$ for some $p, q \in \mathbb{N}$. For example, the set $\{10, 13, 16, 19, \ldots\} = \{10 + 3n : n \in \mathbb{N}\}$ is an arithmetic progression.

(A) Show that if $L \subseteq a^*$ and $\{|w| : w \in L\}$ is an arithmetic progression, then L is regular. (An example of such a language L is the language $\left\{a^{10+3n} : n \in \mathbb{N}\right\}$.)

(B) Use a counterexample to show that if $L \subseteq \{a, b\}^*$ and $\{|w| : w \in L\}$ is an arithmetic progression, then L need not be regular.

(C) (Challenge!! Not required; worth up to 2 extra credit points.) Let $S \subseteq \mathbb{N}$ be an infinite set that does not contain any arithmetic progression as a subset. Let L be an infinite language and $L \subseteq \{w \in \Sigma^* : |w| \in S\}$. (In other words. $\forall w \in L, |w| \in S$.) Prove that L is not regular.

Note: On every problem set we will provide a challenge problem, generally significantly more difficult than the other problems in the set, but worth only a few points. It is recommended that you attempt these problems, but only after completing the rest of the assignment.

PROBLEM 5 (4 points, suggested length of $\frac{1}{4}$ page)

Define an *arithmetic function* as any function $\mathbb{N} \to \mathbb{N}$. Use a diagonalization argument to prove that the set of arithmetic functions is uncountably infinite.

PROBLEM 6 (4+2 points, suggested length of 1/2 page)

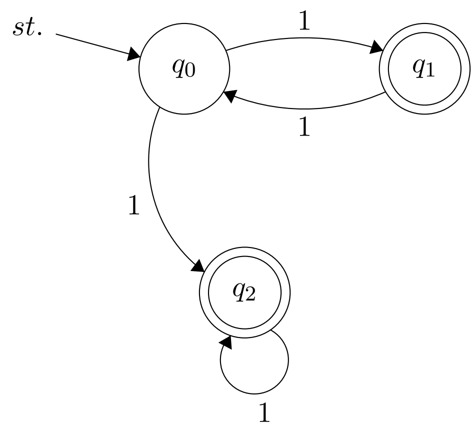
Let Σ be some alphabet. Consider $L = \{R : R \text{ is a regular expression for a language over} \Sigma\}$.

(A) Prove that L is not a regular language.
(B) Give a context-free grammar which generates L.

PROBLEM 7 (6 points, suggested length of 1/2 page)

An *All-Paths-NFA* is exactly the same as an NFA except that it is defined to accept a string x only if *all* computation paths on x end in an accept state and rejects x otherwise. (In contrast, a standard NFA accepted a string if *any* computation path leads to an accepting state). Show that a language L is regular if and only if it is recognized by some All-Paths-NFA.

For example, given the NFA:



The string "11111" would be accepted, while "1111" would be rejected.