

Additional Practice Final #2

This additional practice final is an actual CS103 final exam from a previous quarter. The structure of this exam is not the same as the structure of the upcoming final exam, but the sorts of questions on it are similar to what you might expect to see on the actual exam.

This practice final is **not worth any extra credit points**, but should be a good resource to help you prepare for the exam.

Enjoy!

FINITE AUTOMATA

1. (5 points) Let L be the following language:

$L = \{ w \mid w \in \{0, 1\}^* \text{ and the number of 0's in } w \text{ is divisible by 2 and the number of 1's in } w \text{ is divisible by 3} \}$.

Draw a state diagram for a DFA whose language is L .

2. (5 points)

Draw a state diagram for an NFA over $\{0, 1\}$ that accepts strings that consist of either

01 followed by 01 repeated zero or more times or
010 followed by 010 repeated zero or more times.

For example, acceptable strings would include 01, 010101, and 010010.

REGULAR EXPRESSIONS and REGULAR LANGUAGES

3. (15 points) There are three parts to this question.

Let $A = (Q, \Sigma, \delta, q_0, \{q_f\})$ be an NFA such that there are no transitions into q_0 and no transitions out of q_f . Describe the language accepted by each of the following modifications of A , in terms of $L = L(A)$:

[Your answer may be a regular expression, or an English description, or a combination of the two.]

- (a) The automaton constructed from A by adding an ϵ -transition from q_f to q_0 .
- (b) The automaton constructed from A by adding an ϵ -transition from q_0 to every state reachable from q_0 (along a path whose labels may include symbols from Σ as well as ϵ).
- (c) The automaton constructed from A by adding an ϵ -transition to q_f from every state that can reach q_f along some path.

4. (10 points) Write a regular expression for the set of all strings of 0's and 1's such that every pair of adjacent 0's appears before every pair of adjacent 1's.

[So this means that no 00 appears anywhere after the first 11. It is not saying that the 00's are immediately before the 11's.]

CONTEXT-FREE LANGUAGES

5. (15 points) Recall that a *derivation* for a CFG G is a sequence of substitutions using the rules of the grammar, ending with a string in the language. We will call the number of substitutions the *length* of the derivation.

Example: if, for some grammar, we had the derivation $S \Rightarrow 0S1 \Rightarrow 00S11 \Rightarrow 00211$, the length of the derivation is 3.

Let G be any context-free grammar. Prove that for $n > 0$, the number of strings that have a derivation in G of length n or less is finite.

6. (15 points) Use the Pumping Lemma to show that the following language is not regular. We have laid out the proof for you, so use this form and fill in the blank spots.

$$L = \{wc^n \mid w \in (a, b)^* \text{ and } (\text{the number of a's in } w) = n \text{ or } (\text{the number of b's in } w) = n\}$$

[a, b, and c are all in the alphabet.]

The proof is by contradiction. Assume that L is _____. Let p be the pumping length for L as given by the pumping lemma.

Consider the string $w =$ _____.

Clearly, w belongs to L, because _____.

By the Pumping Lemma, we can write $w = xyz$ where for each $i \geq 0$, $xy^iz \in L$, $|y| > 0$, and $|xy| \leq p$.

(the rest of the proof goes here)

Thus, we have a contradiction, and L is not regular.

TURING MACHINES/UNDECIDABILITY

7. (15 points) Show that the following language is undecidable by reduction from A_{TM} :

$$L_{TM} = \{\langle M, w \rangle \mid \text{Turing machine } M, \text{ when given } w \text{ as input, tries to move its head left from the leftmost tape cell, at some point in the computation}\}$$

NP-COMPLETENESS

8. (20 points) We define the class coNP to be the languages that are the complements of languages in NP . As Sipser points out, we don't know whether coNP is different from NP .

Prove that if $\text{P} = \text{NP}$, then $\text{NP} = \text{coNP}$.