
Problem Set 4

This problem set is due on **Thursday September 29th, by 5:00pm.**

Use the CS172 drop box.

Write **your name and your student ID number** on your solution. Write legibly. The description of your proofs should be as *clear* as possible (which does not mean *long* – in fact, typically, good clear explanations are also short.) Be sure to be familiar with the collaboration policy, and read the overview in the class homepage www.cs.berkeley.edu/~luca/cs172.

Typo in problem 3.b, part (ii) of the definition, corrected on 9/26.

1. Let $\Sigma = \{0, 1\}$ and let L be the set of strings of the form $x \circ 1 \circ y$ where x is an integer n written in binary and y is a sequence of n zeroes. For example 110 and 1101000000 are in L , but 1011 and 10100000 are not.

Prove that L is not regular.

2. Let $\Sigma = \{a, b\}$. Let L be the set of strings of the form $a^n b^m$ where $n > m > 0$. Show that L is not regular.
3. Fix a finite alphabet Σ . Let L be a regular language. Let M be the DFA for L described in the Myhill-Nerode Theorem. This problem will show that any other DFA A for L can be “condensed” down to M . It will also show that if N is another DFA for L with the same number of states, it is just a renaming of M .

- (a) A node t in a DFA D is *reachable* if there is some input x so that D is in state t after input x .

Let A be a DFA which recognizes L . Define A_{reach} to be A restricted down to the reachable nodes of A . Show that A_{reach} is a valid DFA and that A_{reach} also recognizes L .

- (b) Suppose we have two DFAs with the same alphabet, $A = (Q_A, \Sigma, \delta_A, q_{0,A}, F_A)$ and $B = (Q_B, \Sigma, \delta_B, q_{0,B}, F_B)$. If we have some map from the states of A to the states of B , say $f : Q_A \rightarrow Q_B$, we say that f is a *DFA homomorphism* if the following properties are true:

- i. $f(q_{0,A}) = q_{0,B}$
- ii. For any letter $l \in \Sigma$ and any state $s \in Q_A$, $f(\delta_A(s, l)) = \delta_B(f(s), l)$
- iii. A state s is in F_A if and only if $f(s)$ is in F_B .

Intuitively, this means f preserves the structure of the automata; the start state goes to the start state, accepting states go to accepting states, and transitions do not get messed up.

Show that for any DFA A which recognizes L , there is a DFA homomorphism from A_{reach} to M . (In fact, this function is also unique.) **Hint:** Any state s of A_{reach} can be gotten to by some string x . Figure out where that string must go in M . Then say why it doesn't matter which string you picked for x .

This shows that any DFA for L can be turned into M by first removing the unreachable states, and then taking applying a structure preserving function.

- (c) Prove that if f is a DFA homomorphism from any DFA A into M , then f is surjective. I.e. for every state $m \in M$ there is some state $a \in A$ so that $f(a) = m$.
- (d) If A is a DFA for L with the same number of states as M , show that:
 - i. $A = A_{reach}$ (I.e. every node of A is reachable)
 - ii. Let f be the homomorphism from $A = A_{reach}$ to M described in part (b). Prove that f is injective. I.e. if s and t are two distinct nodes in A , then a homomorphism f from A to M must have $f(s) \neq f(t)$.

This means f matches up the states of A and M so that the everything works the same. In other words, A is a renaming of M .