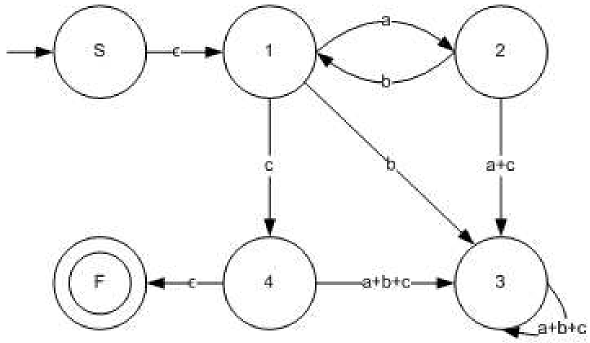
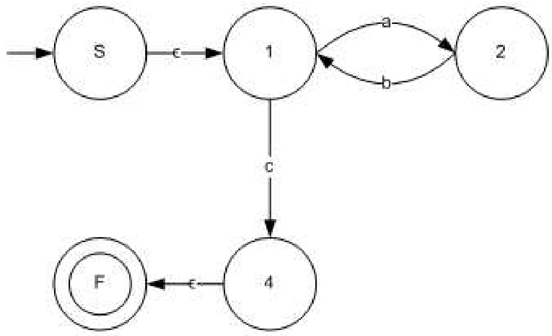
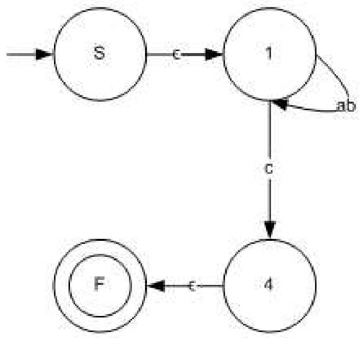
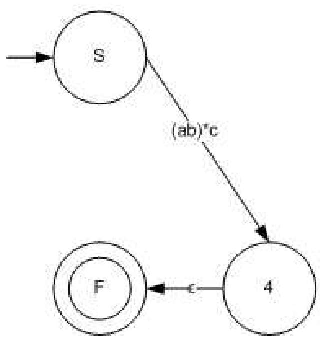
CS 154 Intro. to Automata and Complexity Theory Handout 12
Autumn 2008 David Dill October 13, 2009

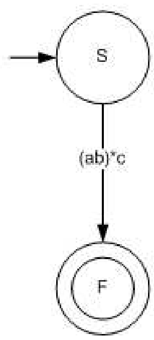
Solution Set 2











Problem 1 $L(M) = (ab)^*c$

Problem 2

a. $L_1 = \{w \mid w$ contains twice as many 1's as 0's$\}$ is not regular.

Pumping Lemma:

Proof Assume L_1 is regular and apply the Pumping Lemma.

By P.L. there exists some pumping constant $n > 0$.

We choose $w = 0^n 1^{2n}$, which is in L_2 and satisfies $|w| \geq n$.

By P.L. $w = xyz$ such that $|xy| \leq n$ and $y \neq \epsilon$. Note that the first condition implies that both x and y contain only 0's. Let $|x| = a$ and $|y| = b$, the second condition implies $b > 0$.

We choose $k = 0$: $xz \in L_1$ by P.L., but $xz = 0^{n-b}1^{2n}$ which is not in L_1 since $b \neq 0$. Thus we get a contradiction, which implies our initial assumption was wrong. We conclude that L_1 is not regular. □

Closure Properties:

Proof Suppose L_1 were regular. Define $h : \{0,1\} \to \{0,1\}$ so that $h(0) = 0$ and $h(1) = 11$. Then $\{0^i 1^i \mid i \geq 0\} = h^{-1}[L_1 \cap L(0^*1^*)]$. To see this, we show that $w \in \{0^i 1^i \mid i \geq 0\}$ iff $h(w) \in h^{-1}[L_1 \cap L(0^*1^*)]$. If $w \in \{0^i 1^i \mid i \geq 0\}$, it is obvious that $h(w) \in L_1 \cap L(0^*1^*)$. If $w \notin \{0^i 1^i \mid i \geq 0\}$, then it must either have different numbers of 0's and 1's, in which case $h(w)$ will not have twice as many 1's and 0's as required to be in L_1, or w has a 1 before a 0, in which case $h(w)$ will have the same property and will not be a member of $L(0^*1^*)$.) Since the regular languages are closed under inverse homomorphisms, $\{0^i 1^i \mid i \geq 0\}$ would be regular, but it is known not to be. Therefore, L_1 must not be regular. □

b. The language $L_2 = \{0^n 1^m 2^{n-m} \mid n \geq m \geq 0\}$ over $\Sigma = \{0, 1, 2\}$.

Pumping Lemma proof:

Proof Assume L_2 is regular and apply the Pumping Lemma.

By P.L. there exists some pumping constant $n > 0$.

We choose $w = 0^n 1^n$, which is in L_2 and satisfies $|w| \geq n$.

By P.L. $w = xyz$ such that $|xy| \leq n$ and $y \neq \epsilon$. Note that the first condition implies that both x and y contain only 0's. Let $|x| = a$ and $|y| = b$, the second condition implies $b > 0$.

We choose $k = 2$: $xy^2z \in L_1$ by P.L., but $xy^2z = 0^{n+b}1^n$ which is not in L_2 since xy^2z needs b 2s at the end to be in L_2, and $b > 0$. Thus we get a contradiction, which implies our initial assumption was wrong. We conclude that L_2 is not regular. □

Closure Properties proof:

Proof

Suppose $L_2 = \{0^n 1^m 2^{n-m} \mid n \geq m \geq 0\}$ were regular.

Now, consider the following homomorphism from the alphabet $\Sigma_0 = \{0, 1, 2\}$ to the alphabet $\Sigma_1 = \{0, 1\}$: $h(0) = 0, h(1) = 1, h(2) = 1$, so $\{0^i 1^i \mid i \geq 0\} = h(L_2)$. But the regular languages are closed under homomorphisms and this language is not regular, so L_2 must not be regular. □

Problem 3 $L = \{0^c \mid c \text{ is a perfect cube}\}$ is not regular.

Proof Assume L is regular and apply the Pumping Lemma.

By P.L. there exists some pumping constant $n > 0$.

We choose $w = 0^{n^3}$ which is in L and also satisfies $|w| \geq n$ since $n^3 \geq n$ for all $n > 0$.

By P.L. $w = xyz$ such that $|xy| \leq n$ and $y \neq \epsilon$. Note that all x, y, and z contain only 0's. Let $|x| = a$ and $|y| = b$, the conditions imply $0 < b \leq n$.

We choose $k = 2$: $xy^2z \in L$ by P.L., but $xy^2z = 0^{n^3+b}$. But $n^3 + b$ cannot be a perfect cube since the bounds on b imply the following:

$$n^3 < n^3 + b \leq n^3 + n < n^3 + 3n^2 + 3n + 1 = (n+1)^3$$

So $n^3 + b$ is strictly between the values of two *consecutive* perfect cubes, and cannot be a perfect cube itself. Thus we get a contradiction, which implies our initial assumption was wrong. We conclude that L is not regular. □

Problem 4 Given a DFA $D = (Q, \Sigma, \delta, q_0, F)$ that accepts L, we can construct a new DFA $D' = (Q, \Sigma, \delta, q_0, F')$ that has the same set of states Q, alphabet Σ, transition function δ, and initial state q_0 as the original DFA D. The set of final states F' for D' is defined as follows:

$$F' = \{q' \in Q \mid \exists q \in F, x \in \Sigma^* \text{ such that } \hat{\delta}(q', x) = q\}$$

In other words, a state q' is a final state of D' if there is a path from q' to a final state of D.

We will show that D' accepts $init(L)$. The proof has two parts: (1) all strings accepted by D' are in $init(L)$, and (2) all strings in $init(L)$ are accepted by D'.

Part 1. Suppose that D' accepts a string w, terminating in final state $q' \in F'$ (i.e., $\hat{\delta}(q_0, w) = q'$). By construction, there exists a path from q' to some state $q \in F$. Let x be the string defined by such a path, i.e. $\hat{\delta}(q', x) = q$. Then the string wx is accepted by D, since $\hat{\delta}(q_0, wx) = \hat{\delta}(\hat{\delta}(q_0, w), x) = \hat{\delta}(q', x) = q \in F$. Thus all strings accepted by D' are in $init(L)$.

Part 2. Consider a string $wx \in L$. We will show that D' accepts w by showing that $q' = \hat{\delta}(q_0, w) \in F'$. Since $wx \in L$, we know that $q = \hat{\delta}(q_0, wx) \in F$. Using the equivalence $q = \hat{\delta}(q_0, wx) = \hat{\delta}(\hat{\delta}(q_0, w), x) = \hat{\delta}(q', x)$, we can conclude that $\hat{\delta}(q', x) \in F$, meaning that $q' \in F'$ by the definition of F'. Thus all strings in $init(L)$ are accepted by D'.

Problem 5 Let L' be the language of the regular expression 1^*02^*. We claim that $h^{-1}(L) = L'$. To establish this claim, we must show that $w \in L'$ if and only if $w \in h^{-1}(L)$. By the definition of inverse homormorphisms, $w \in h^{-1}(L)$ if and only if $h(w) \in L$. Thus, we must show that $w \in L'$ if and only if $h(w) \in L$.

First, consider a string $w = 1^i 0 2^j \in L'$, where $i, j \geq 0$. Applying the homomorphism h to w, we obtain $h(w) = (ab)^i a(ba)^j$, which can be rewritten as $h(w) = a(ba)^{(i+j)}$. Hence, $h(w) \in L$.

Now, consider a string $w \notin L'$. Since w is not in the language of the regular expression above, at least one of the following conditions must hold. We show that if any of the conditions holds, then $h(w) \notin L$.

1. If w contains no 0's, then $|h(w)|$ is even, so $h(w)$ cannot be in L, which only contains strings of odd length.

2. If w contains at least two occurrences of 0, then w has a substring $0x0$, where $|x| \geq 0$ and x consists entirely of 1's and 2's. Applying h to this substring, we conclude that $h(w)$ contains the substring $a \cdot h(x) \cdot a$, where $|h(x)|$ is even. Since any string containing two a's that are separated by an even number of symbols is not in L, we have $h(w) \notin L$.

3. If w contains a 0 before a 1, then it has a substring $0x1$, where $|x| \geq 0$ and x consists entirely of 2's. So, $h(w)$ contains the substring $a \cdot h(x) \cdot (ab)$, where $|h(x)|$ is even, and again there are two a's in $h(w)$ separated by an even number of symbols.

4. Similarly, if w contains a 2 before a 0, then it has a substring $2x0$, where $|x| \geq 0$ and x consists entirely of 1's. In this case, $h(w)$ contains the substring $(ba) \cdot h(x) \cdot a$, where $|h(x)|$ is even, and again $h(w) \notin L$.

This argument shows that $w \in L'$ if and only if $h(w) \in L$, and therefore establishes the claim that $h^{-1}(L) = L'$.