CLEMSON UNIVERSITY, SCHOOL OF COMPUTING
CPSC 3500 FOUNDATIONS OF COMPUTER SCIENCE

Assignment 4: Finite Automata

Return by 11:59pm 10/2/2018

Problem 1 10%

Let L be the language $AnBn = \{a^n b^n | n \geq 0\}$.

(a) Find two distinct strings $x, y \in \{a, b\}^*$ that are not L-distinguishable.

(b) Find an infinite set of pairwise L-distinguishable strings.

(c) Can you draw FA for $AnBn$?

Problem 2 10%

Let n be a positive integer and $L = \{x \in \{a, b\}^* | \ |x| = n$ and $n_a(x) = n_b(x)\}$. What is the minimum number of states in any FA that accepts L? Give reasons for your answer.

Problem 3 20%

Choose any 4 items out of (a)-(h). For each of the following languages $L \subseteq \{a, b\}^*$, show that the elements of the infinite set $\{a^n | n \geq 0\}$ are pairwise L-distinguishable.

(a) $L = \{a^n b a^{2n} | n \geq 0\}$

(b) $L = \{a^i b^j a^k | k > i + j\}$

(c) $L = \{a^i b^j | j = i$ or $j = 2i\}$

(d) $L = \{a^i b^j | j$ is a multiple of $i\}$

(e) $L = \{x \in \{a, b\}^* | n_a(x) < 2n_b(x)\}$

(f) $L = \{x \in \{a, b\}^* |$ no prefix of x has more b's than a's$\}$

(g) $L = \{a^{n^3} | n \geq 1\}$

(h) $L = \{ww | w \in \{a, b\}^*\}$

Problem 4 10%

For each of the languages in Problems 3c, and 3e, use the pumping lemma to show that it cannot be accepted by an FA. (I recommend to show this for all languages in the previous problem but don't submit all of them.)

Problem 5 10%

Let n be a positive integer, and let L be the set of all strings in Pal of length $2n$. In other words,

$$L = \{xr(x) | x \in \{a,b\}^n\},$$

where $r(x)$ is the string reverse function. What is the minimum number of states in any FA that accepts L? Give reasons for your answer.

Problem 6 10%

Suppose L is a language over $\{a,b\}$, and there is a fixed integer k such that for every $x \in \Sigma^*$, $xz \in L$ for some string z with $|z| \leq k$. Does it follow that there is an FA accepting L? Why or why not?

Problem 7 20%

Choose any 4 items out of (a)-(j). For each statement below, decide whether it is true or false. If it is true, prove it. If it is not true, give a counterexample. If you prove it, you can use any theorem we saw in class. All parts refer to languages over the alphabet $\{a,b\}$.

(a) If $L_1 \subseteq L_2$, and L_1 cannot be accepted by an FA, then L_2 cannot.

(b) If $L_1 \subseteq L_2$, and L_2 cannot be accepted by an FA, then L_1 cannot.

(c) If neither L_1 nor L_2 can be accepted by an FA, then $L_1 \cup L_2$ cannot.

(d) If neither L_1 nor L_2 can be accepted by an FA, then $L_1 \cap L_2$ cannot.

(e) If L cannot be accepted by an FA, then L' cannot (Reminder: L' is the complement to L).

(f) If L_1 can be accepted by an FA and L_2 cannot, then $L_1 \cup L_2$ cannot.

(g) If L_1 can be accepted by an FA, L_2 cannot, and $L_1 \cap L_2$ can, then $L_1 \cup L_2$ cannot.

(h) If L_1 can be accepted by an FA and neither L_2 nor $L_1 \cap L_2$ can, then $L_1 \cup L_2$ cannot.

(i) If each of the languages $L_1, L_2, \ldots$ can be accepted by an FA, then $\cup_{n=1}^{\infty} L_n$ can.

(j) If none of the languages $L_1, L_2, \ldots$ can be accepted by an FA, and $L_i \subseteq L_{i+1}$ for each i, then $\cup_{n=1}^{\infty} L_n$ cannot be accepted by an FA.

Problem 8 10%

A set S of nonnegative integers is an arithmetic progression if for some integers n and p,

$$S = \{n + ip | i \geq 0\}.$$

Let A be a subset of $\{a\}^*$, and let $S = \{|x| \mid x \in A\}$. Show that if S is an arithmetic progression, then A can be accepted by an FA.

Solutions to Assignment 4

Solution to Problem 1

(a) Example: b, and ba. Neither can lead to an element in L.

(b) $\{a^n | n \geq 0\}$ are all pairwise indistinguishable

(c) There is no such FA.

Solution to Problem 2

Sketch: For L to be nonempty, n must be even. If $n = 2m$, there is an FA with $(m+1)^2 + 1$ states. It has a state for each of the ordered pairs (i, j), where $0 \leq i \leq m$, and $0 \leq j \leq m$, and i, j represent the numbers of a's, and b's. For such a pair (i, j), each string is a prefix for an element of L. There is one additional state N corresponding to all the nonprefixes of elements of L. The transactions are of type $(i, j) \to (i+1, j)$, and $(i, j) \to (i, j+1)$. All illegal transactions will lead to N.
This is the minimum possible number of states, because $(m+1)^2 + 1$ strings obtained by choosing one for each state are pairwise L-distinguishable.
Proof by induction on n is also possible.

Solution to Problem 3

Assume in each part that $0 \leq i < j$. The string(s) that distinguishes a^i, and a^j is (are)

(a) ba^{2i}

(b) see answer in the textbook

(c) b^{2j}

(d) ab^{i+1}

(e) if $i = 2p+1$ then b^{p+1}, otherwise ab^{p+1}

(f) b^j

(g) For each integer m, so that $m^3 \geq i$, $a^i a^{m^3-i} \in L$, and $a^j a^{m^3-i} = a^{m^3+(j-i)}$. If m is a number large enough that $m^3 + (j-i) < (m+1)^3$, then a^{m^3-i} distinguishes a^i, and a^j.

(h) $ba^i b$

Solution to Problem 4

(a) Suppose there exists FA M, let n be the integer in the pumping lemma, and let $x = a^n b^n$. Clearly, $|x| \geq n$. Then $x = uvw$ for some u, v, w that satisfy all three conditions. The first condition says that $|uv| \leq n$, i.e., v must be a^j for some $j > 0$. Therefore, $uv^2 w = a^{n+j} b^n$, and this string cannot be in L because n is neither $n+j$ nor $2(n+j)$. This is the necessary contradiction.

(b) Similar to (c).

(c) Take $x = a^{2n-1} b^n$. Then $x = uvw$ for some u, v, w satisfying all three conditions. As before v must be a^j for some $j > 0$, and $uv^2 w = a^{2n+j-1} b^n$. Since $j - 1 \geq 0$, $2n + j - 1 \geq 2n$, and this is a contradiction.

Solution to Problem 6
No. Let L be the language of nonpalindromes over $\{a, b\}$. L cannot be accepted by an FA because its complement cannot. However, if x begins with a, $xb \in L$; if x begins with b, $xa \in L$, and if $x = \Lambda$, $xab \in L$. Therefore, L satisfies the condition for $k = 2$.

Solution to Problem 7

(a) False. Every language is a subset of $\{a, b\}^*$ which can be accepted by FA.

(b) False. A counter-example is a finite subset of a language that cannot be accepted by FA.

(c) False. If $L \subseteq \Sigma^*$ cannot be accepted by an FA, then L' cannot but $L \cup L' = \Sigma^*$.

(d) False. If $L \subseteq \Sigma^*$ cannot be accepted by an FA, then L' cannot but $L \cap L' = \emptyset$.

(e) True

(f) False. L_2 could be a subset of L_1, for example.

(g) True. If $L_1 \cap L_2$ can be accepted by FA, so can $L_1 - (L_1 \cap L_2) = L_1 - L_2$. Now $L_2 = (L_1 \cup L_2) - (L_1 - L_2)$. Therefore, if $L_1 \cup L_2$ could be accepted by an FA, L_2 could be also.

(h) False. L_1 can be Σ^*.

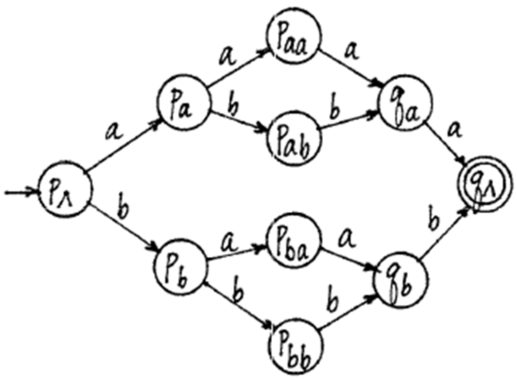
(i) False. Every language is a union of one-element languages.

(j) False.

Solution to Problem 8
Suppose $A = \{a^{n+ip} | i \geq 0\}$. A finite automaton accepting A can be constructed as follows. Because the alphabet is $\{a\}$, only one transition is needed from each state. If the initial state is q_0 , there are additional distinct states $q_1, q_2, ..., q_n$ and transitions from q_j to q_{i+1} for $0 \leq j < n$. If $n > 0$, the state q_n is the only one of these states that is accepting. If $p > 0$, there are nonaccepting states $q_{n+1}, q_{n+2}, ..., q_{n+p-1}$, a transition from q_j to q_{j+1} for $n \leq j < n + p - 1$, and a transition from q_{n+p-1} back to q_n.

Solution to Problem 5

Here is a diagram of an FA accepting L in the case $n = 2$, which we call M_2.



We have simplified the picture by leaving out one of the states. Every transition not shown explicitly is assumed to go to a nonaccepting state, from which both transitions return to that state.

It is straightforward to generalize this to arbitrary n, so as to obtain an FA M_n. For each string x of length $\leq n$, there is a state p_x corresponding to x (that is, for which $\{y \mid \delta^*(q_0, y) = p_x\}$ is $\{x\}$. In addition, for each x with $|x| < n$, there is another state, which we call q_x, corresponding to the set $\{y \mid yx \in L\}$. The states p_x and q_x account collectively for all the strings that are prefixes of elements of L, and all the others are rejected. Since there are $2^{n+1} - 1$ strings of length $\leq n$ and $2^n - 1$ strings of length $< n$, the total number of states in the FA is $(2^{n+1} - 1) + (2^n - 1) + 1 = 3 * 2^n - 1$. Now we show that no FA with fewer states can accept L, by showing that two strings corresponding to different states of M_n are distinguishable with respect to L.

Clearly any string that is a prefix of an element of L is distinguishable from any string that is not. Also, if x and y are both prefixes of elements of L and $|x| \neq |y|$, then x and y are distinguishable. It is therefore sufficient to show: (i) if $|x_1| = |x_2| \leq n$ and $x_1 \neq x_2$, then x_1 and x_2 are distinguishable; and (ii) if $|y_1| = |y_2|$, $y_1 \neq y_2$, $x_1y_1 \in L$, and $x_2y_2 \in L$, then x_1 and x_2 are distinguishable. Statement (i) is easy:

$$x_1 a^{2n-2|x_1|} x_1^r \in L \text{ and } x_2 a^{2n-2|x_1|} x_1^r \notin L$$

Statement (ii) is just as easy: if $x_1y_1 \in L$ and $y_1 \neq y_2$, then since $x_2y_2 \in L$, we must have $x_2y_1 \notin L$. Therefore, y_1 distinguishes x_1 and x_2.