American University of Beirut

CMPS 257

Faculty of Arts & Sciences
Department of Computer Science
Theory of Computation
Fall 2015–16
Saturday, Oct 10th, 2015

Family Name			
First Name			
Id. Num		**Code**	
Section	1. 9:30-10:45 TH 2. 2:00-3:15 TH (Circle one)		
Major			

QUIZ 1 **Duration: 60 Minutes**

Problem	Grade	Your Grade
1	20	
2	20	
3	15	
4	10	
5	10	
6	15	
7	20	
Total	**110**	

Instructions

Write your name, Id number, and your major in the space provided above.

THE EXAM IS OPEN BOOK, BUT "CLOSED NEIGHBOR"!!

Your answers must be presented on the question sheet itself. Your handwriting should be readable so it can be graded. Use the provided space for your answers. If the space is not enough, then most probably you are writing more than what is expected. In case you need more space, put an "explicit pointer" to where your answer is written or continued (e.g. draw an arrow, or say back of page or both !!!…)

There are 7 problems altogether and 7 pages (besides this front page). You may use the back of the sheets for scratch work.

DO NOT CUT OFF ANY PAGE !!

Problem 1 (20 Points)

Let $\Sigma = \{0,1\}$

a. (10) Give a language L such that $\{\varepsilon, 0, 01\}\, L = \{0, 01, 10, 010, 011, 0110\} = B$

Seeking a language L such that $A\,L = B$, where $A = \{\varepsilon, 0, 01\}$ and $B = \{0, 01, 10, 010, 011, 0110\}$

Impossible to find such an L :

$\varepsilon \notin B$, so $\varepsilon \notin L$ (since if $\varepsilon \in L$ then $\varepsilon \in B$, because $\varepsilon \in A$).

So $0 \in L$ (to get $0 \in B$, and since $\varepsilon \notin L$, the only way is to concatenate $\varepsilon \in A$ with 0 in L)

But if $0 \in L$, then $00 \in B$ (concatenation of $0 \in A$ with $0 \in L$) which is not the case.

N. B.

If $B = \{1, 01, 10, 010, 011, 0110\}$, then $L = \{1, 10\}$

b. (10) Show that for any languages A, B $(AB)^*A = A(BA)^*$

Let $w \in (AB)^*A$. Then $w = ux$, where $u \in (AB)^*$ and $x \in A$ **.** So $u = u_1\, u_2 \ldots u_k$ for some $k \geq 0$**,** with each $u_j \in AB$, $j=1,2,\ldots,k$**.**

Now $u_j \in AB$, *means that* $u_j = x_j\, y_j$, $x_j \in A$ and $y_j \in B$, $j=1,2,\ldots,k$.

Thus $u = x_1\, y_1\, x_2\, y_2 \ldots x_k\, y_k$

Where we have dropped the parenthesis, because concatenation is associative. Also, by associativity,

So, $w = (x_1\, y_1)(x_2\, y_2) \ldots (x_k\, y_k)\, x = x_1\, (y_1 x_2)(y_2\, x_3) \ldots (y_k\, x) \in A(BA)^*$

because $x_1 \in A$ and $(y_1 x_2)(y_2\, x_3) \ldots (y_k\, x) \in (BA)^*$.

So $(AB)^*A \subseteq A(BA)^*$.

Similarly, $A(BA)^* \subseteq (AB)^*A$.

Problem 2 (20 Points)

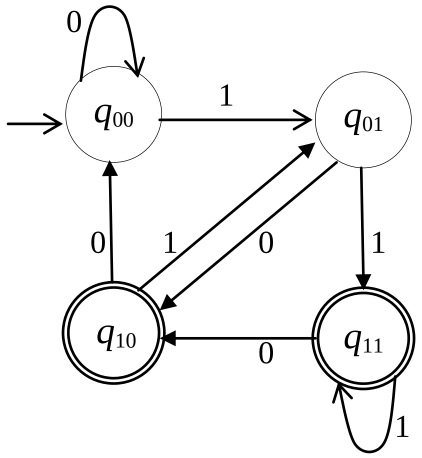
Let $\Sigma = \{0,1\}$, and consider the following language:

$A = \{ w \in \Sigma^* \mid w$ contains a 1 in its second position from the right $\}$

So $0010 \in A$, $11 \in A$, $110 \in A$, *but* 0, 000, 100, 1001 are not in A.

a. (8) Show that A is regular by giving the state diagram of a *deterministic finite automata (DFA)*, with as small a number of states as possible, that recognizes A. (Hint: DFA remembers what are the last two characters read from input)

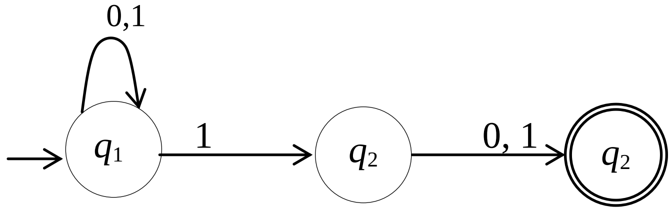
To design the DFA, pretend that the DFA is reading a string of 0's and 1's. Then there are 4 possibilities for the last two characters read: 00, 01, 10, 11. The DFA will have a state for each of these cases. The transition rules, are constructed accordingly. The DFA will be as follows:



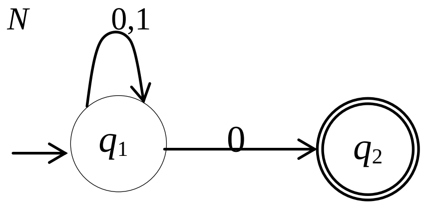
b. (6) Give as simple a regular expression as possible that describes A

$(0 \cup 1)^*1(0 \cup 1)$ or $\Sigma^* 1 \Sigma$

c. (6) Give the state diagram of a *non-deterministic finite automaton (NFA)*, with as small a number of states as possible, that recognizes A.



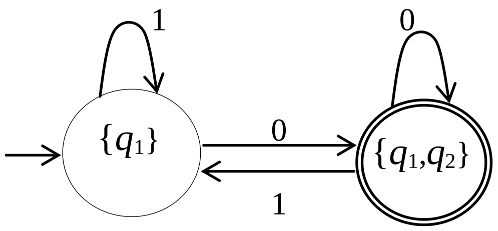
Problem 3   (15 Points)
This problem relates to the "equivalence construction" for constructing the equivalent DFA to a given NFA. Let $\Sigma = \{0,1\}$, and consider the following NFA N:



a. (5) What is the language recognized by N?

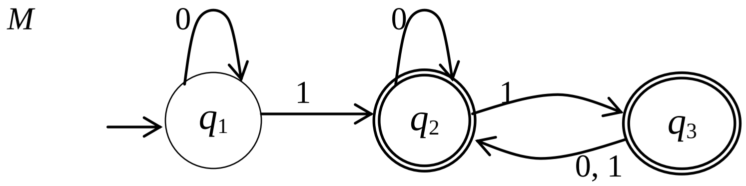
   $L(N) = \{\ w \in \Sigma^* \mid\ \ u0,\ u \in \Sigma^*\ \}$; i.e. set of strings that end with a 0.

b. (10) Give the construction of the DFA that is equivalent to the NFA N above, according to the equivalence construction, where, starting with the start state, only states reachable from the start state are considered. (Should be simple).

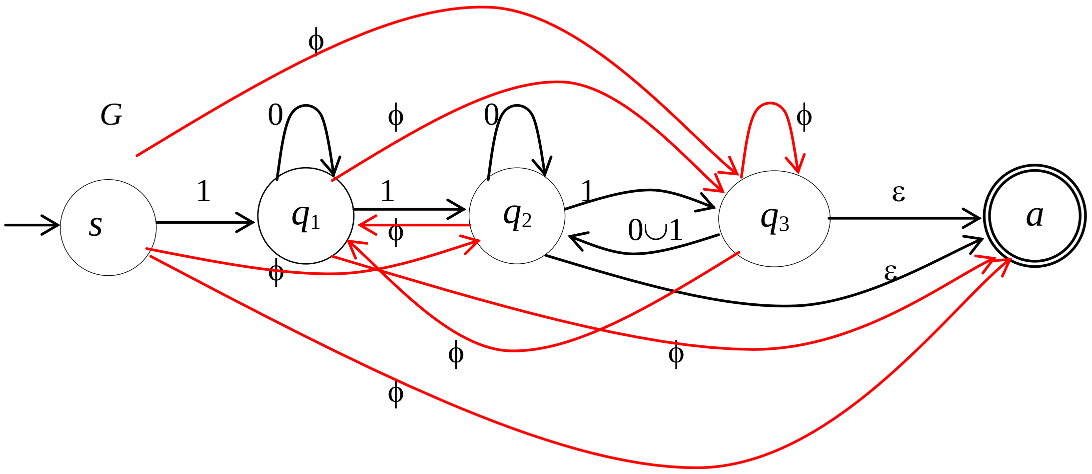


Problem 4 (10 Points)

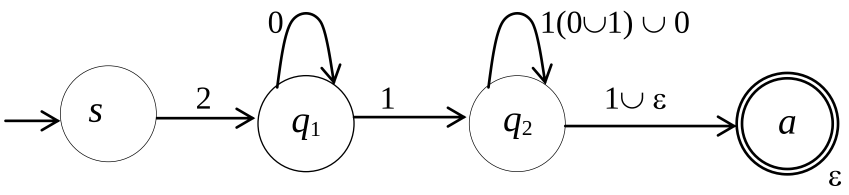
This problem relates to the proof that every regular language can be represented by a regular expression. Consider the DFA M whose state diagram is



a. (4) Give the state diagram of a 5-state GNFA that is equivalent to the given DFA above. (Show all transition arrows, including those labeled with ϕ)



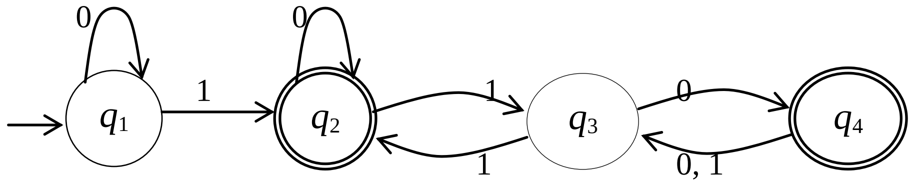
b. (6) Give the state diagram of a 4-state GNFA that is equivalent to the GNFA in (a) obtained by eliminating q_3,



Problem 5 (10 Points)

Consider *M* to be:

M



Here we are going to illustrate the proof of the pumping lemma, as we trace the computation of *M* on the string s = 1101011.Complete the following:

a. (2) The pumping length is __4___, because *M has 4 states*

b. (3) The computation of *M* on s = 1101011, goes through the following sequence of states: (fill out the sequence of states by indicating the subscript.)

$$q_1 \xrightarrow{1} q_2 \xrightarrow{1} q_3 \xrightarrow{0} q_4 \xrightarrow{1} q_3 \xrightarrow{0} q_4 \xrightarrow{1} q_3 \xrightarrow{1} q_2$$

c. (5) So according to the pumping lemma proof, $s = x\,y\,z$ where

x = 11,

y = 01, *and*

z = 011

Problem 6 (20 Points)

Let $\Sigma = \{0,1\}$, and consider the language $D = \{0^n 10^n \mid n \geq 0\}$.

a. (5) In an attempt to show that D is not regular using the pumping lemma, a "proof" starts as follows: "Assume, that D is regular and let p>0 be the pumping length. Let $s=0^p 1^p$. Then $|s| = 2p > p$. So s may be decomposed as $s=xyz$, ect…" This cannot lead to a proof. Why not?

   The string $s=0^p 1^p$ is not "a good choice", as it is not in the language D.

b. (10) Prove that D is not regular using the Pumping Lemma.

   Assume, that D is regular and let p>0 be the pumping length. Let $s=0^p 10^p$. Then $|s| = 2p+1 > p$. So, according to the pumping lemma, s may be decomposed as $s=xyz$, such that:

   1. $xy^iz \in A$, for each $i \geq 0$,
   2. $|y| > 0$; i.e. $y \neq \varepsilon$, and
   3. $|xy| \leq p$.

   We show that there is a contradiction:
   By (3) above, $y=0^l$ where $0 \leq l \leq p$. By (2), $l > 0$. So $0 < l \leq p$.

   But then, $xy^2z = xyyz = 0^{p+l} 10^p$ which is not in D, since $p+l > p$, which contradicts (1) with $i=2$.

   Thus our assumption is false, and D is not regular.

c. (5) Using (b) and closure properties of regular languages, show that $PAL=\{w = w^R \mid w \in \Sigma^*\}$the set of palindromes is not regular.

   It is not difficult to see that

   $$D = PAL \cap 0^*10^*$$

   D is not regular (proven in part (a))
   0*10* represents a regular language (being a regular expression !!!)
   Hence, by closure properties, PAL is not regular.

Problem 7 (15 Points)

Let $\Sigma = \{0,1$. Let A be a language over Σ and let $B = \{ w \in \Sigma^* \mid w = 01u$, for some $u \in A\}$, i.e., a string is in B if it starts with a 01 followed by a string in A.

a. (5) Show that if A is regular then B is regular.

$B = \{01\} \circ A$

{01} is regular, and A is regular, so their concatenation is regular. Thus B is regular

b. (10) Show that if B is regular then A is regular (Hint: Start with a DFA M that recognizes B. Construct a finite automaton that recognizes A).

Let $M(K, \Sigma, \delta, s, F)$ be a DFA that recognizes B. Let p be the state we arrive at in M after processing the character 0 then the character 1; i.e. $q = \delta(s,0)$, and $p = \delta(q,0)$. Define a DFA M' $(K', \Sigma, \delta', s', F')$ where it is the same as M except that its start state is p. i.e.

$K' = K$
$s' = p$
$\delta' = \delta$
$F' = F$

Then M' recognizes A.