CS 321 Theory of Computation

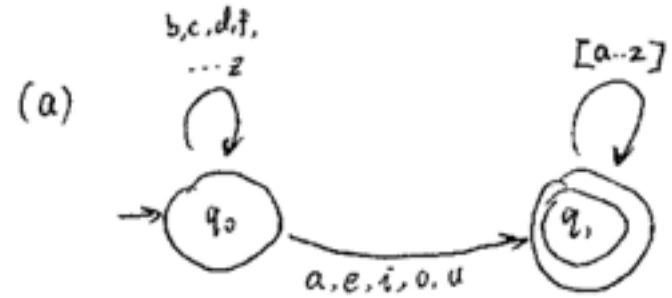
1st Recitation (hw1 solutions)

Dezhong Deng

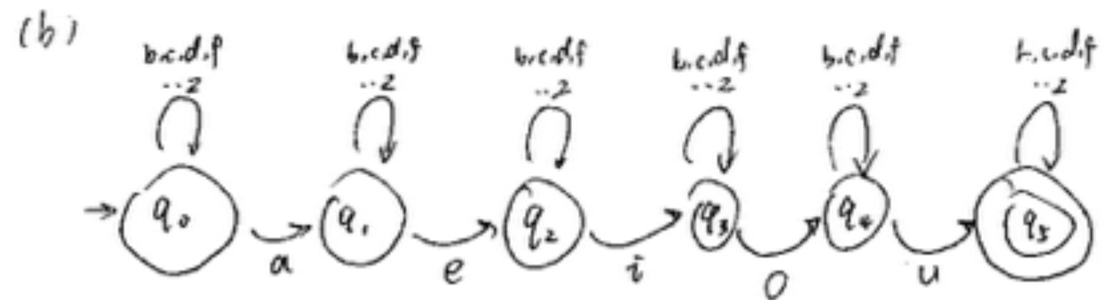
http://classes.engr.oregonstate.edu/eecs/fall2015/cs321/schedule.html

1. Construct a (deterministic) finite automaton for each of the following language. No need to draw trap states.

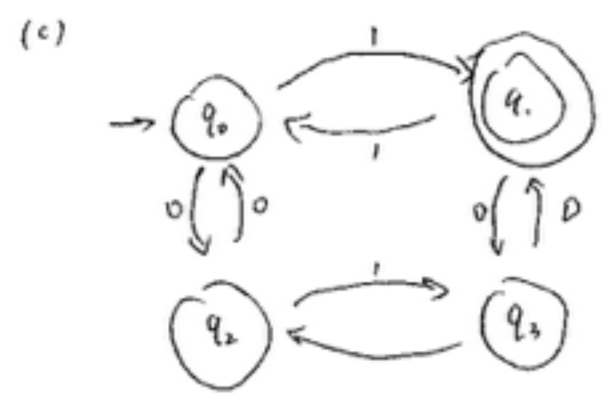
(a) all letter strings with at least a vowel



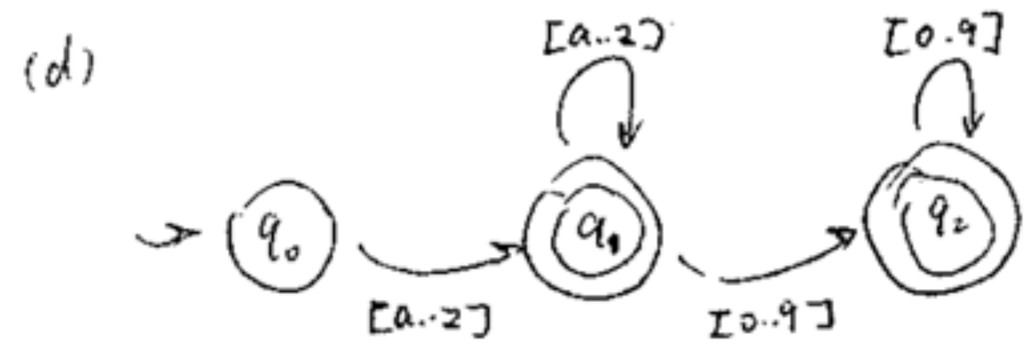
(b) all letter strings with vowels in order
(i.e., each of the five vowels appear once and only once, and in order)



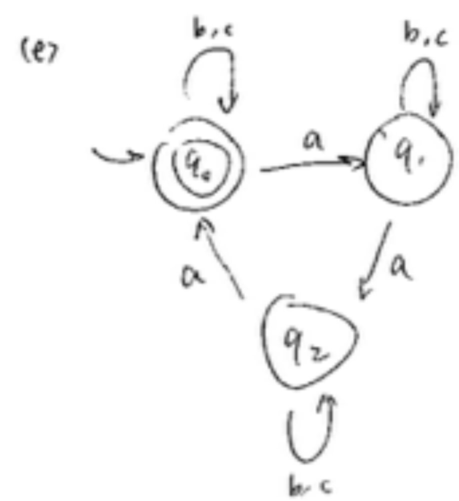
(c) all bitstrings with even numbers of 0s and odd numbers of 1s



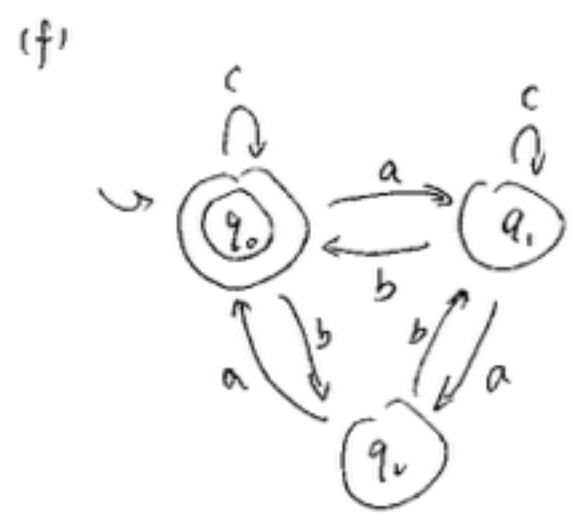
(d) all alphanumeric strings that start with one or more letters followed by zero or more numbers.



(e) all strings over {a,b,c} where the number of a's is divisible by 3



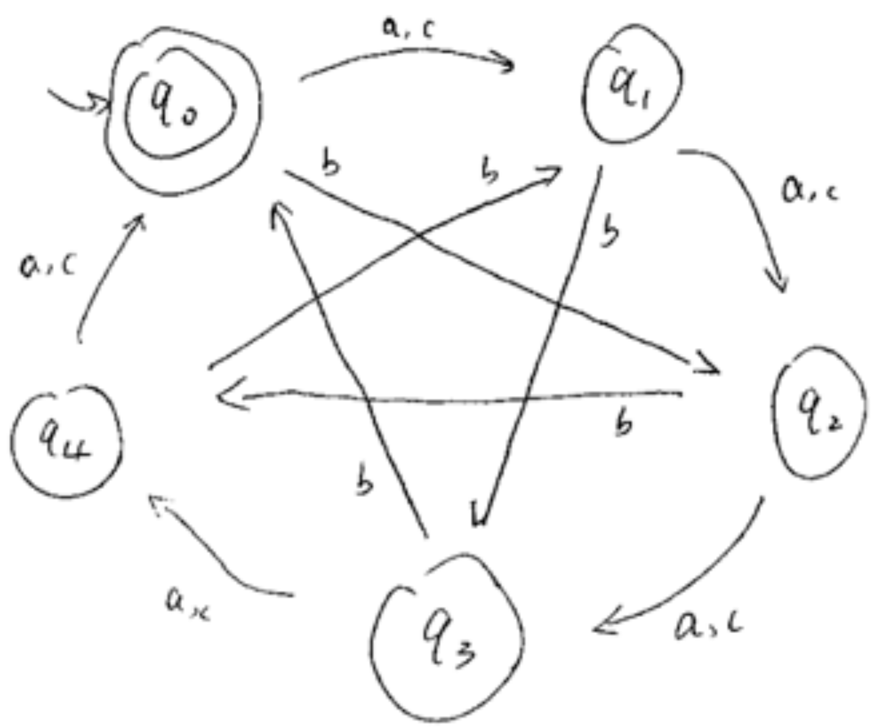
(f) all strings over {a,b,c} where the number of a's minus the number of b's is divisible by 3



(g) all strings over {a,b,c} where the number of a's plus twice the number of b's plus the number of c's is divisible by 5.

what's your general strategy of solving (e-g)?

(g)

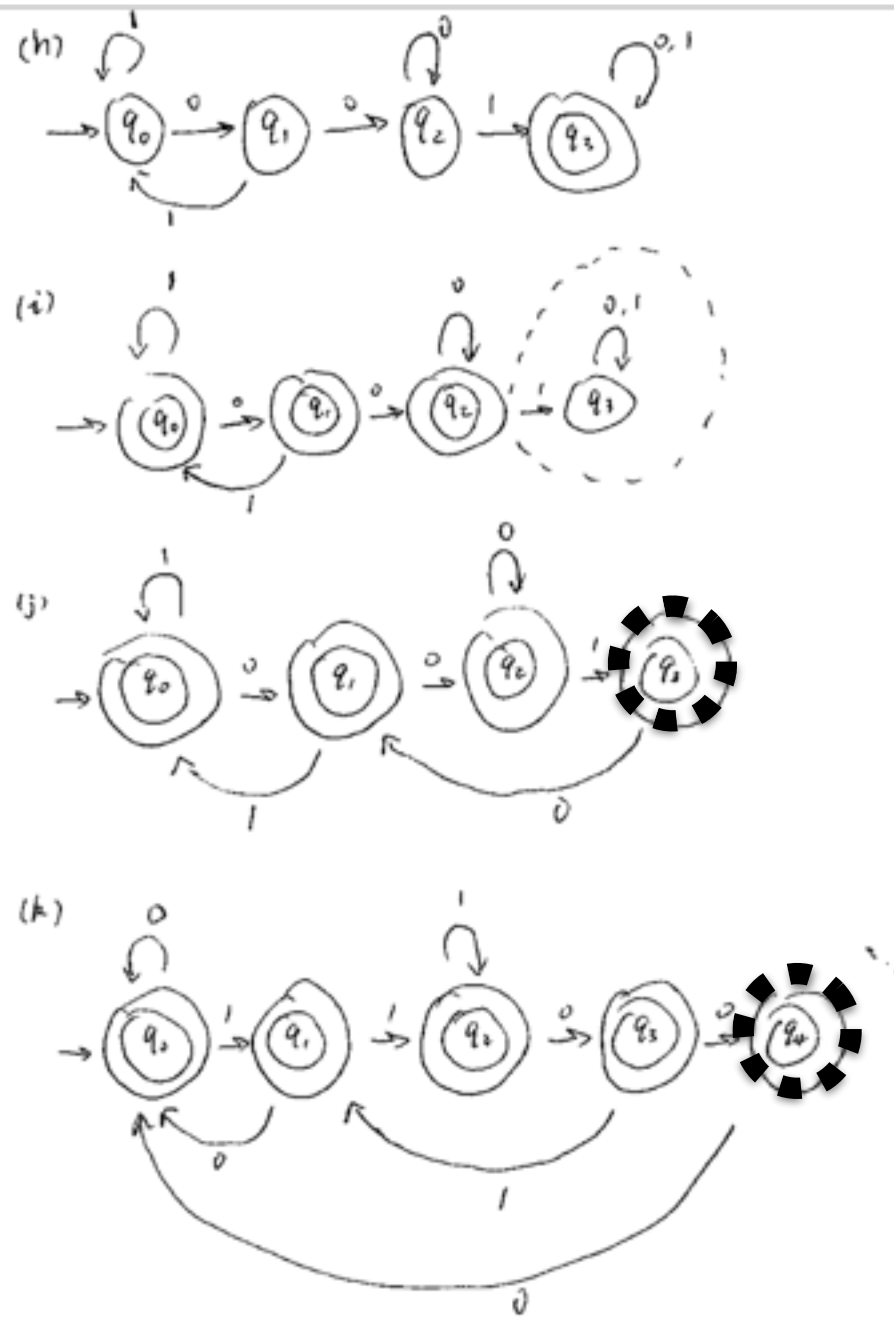


(h) all bitstrings that does contain 001

(i) all bitstrings that does not contain 001

(j) all bitstrings that does not contain 0011

(k) all bitstrings that does not contain 11001



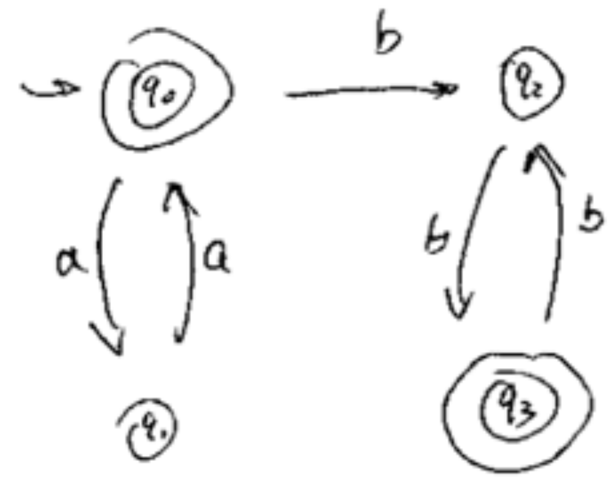
KMP algorithm

(l) all strings over {a,b} where all a's come before any b's,
and the numbers of a's and b's are both even.

(m) all strings over {a,b} where the number of a's equal the number of b's.
can you do this? if not, explain why.

(n) all strings over {a,b} that do not end with ab.

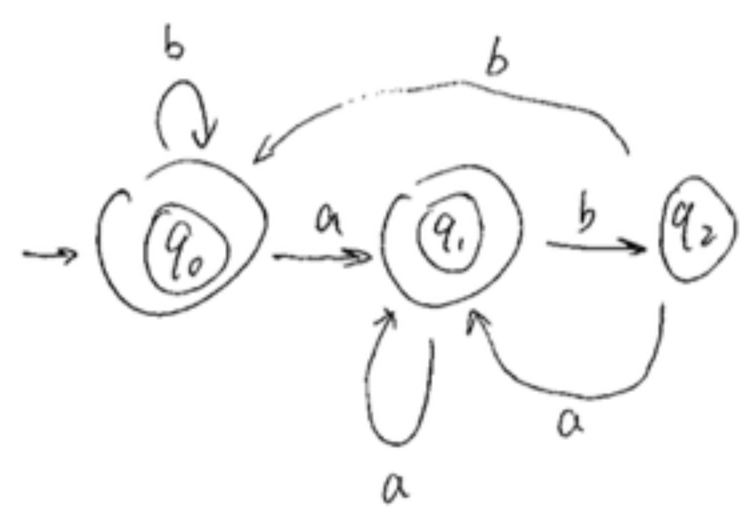
(l)



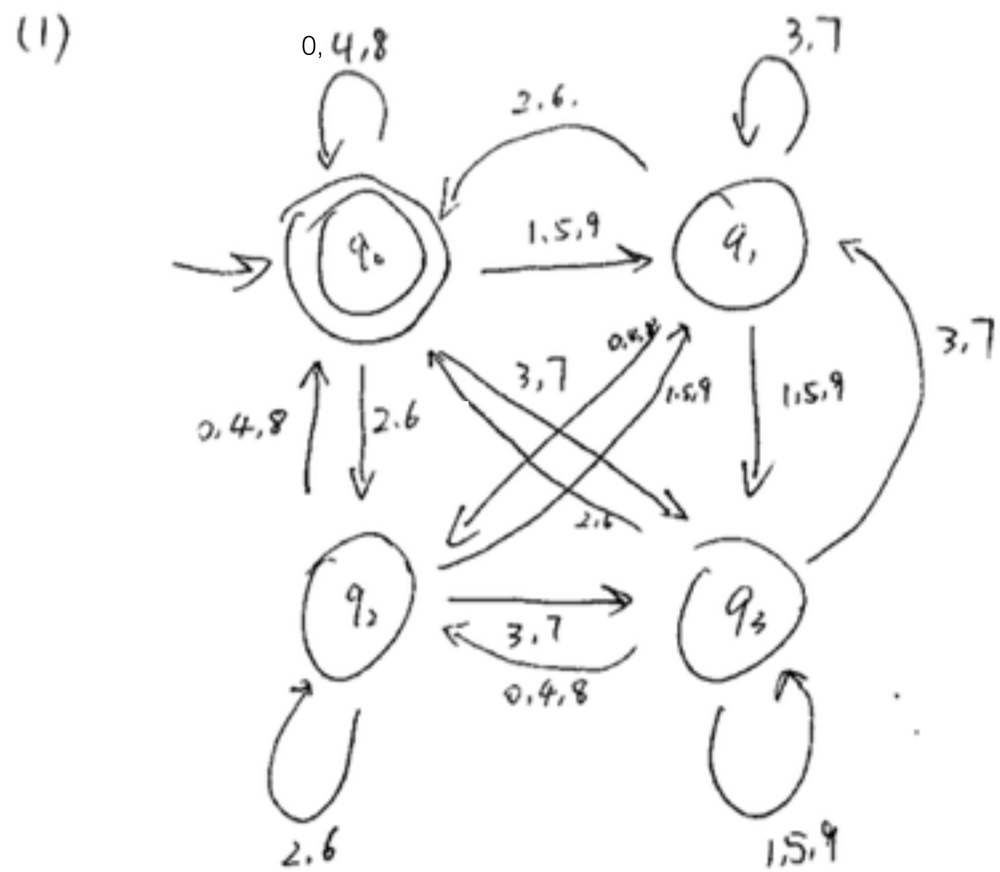
(m) No.

Infinite num of states required ∀

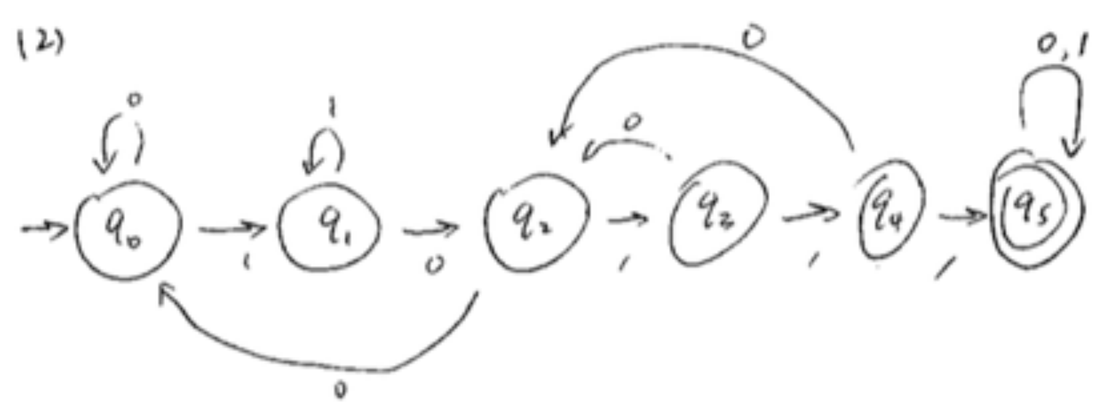
(n)



two more questions: 1) decimal numbers divisible by 4
2) all bit strings that contain 10111



hint: you can use 3 states to solve this problem



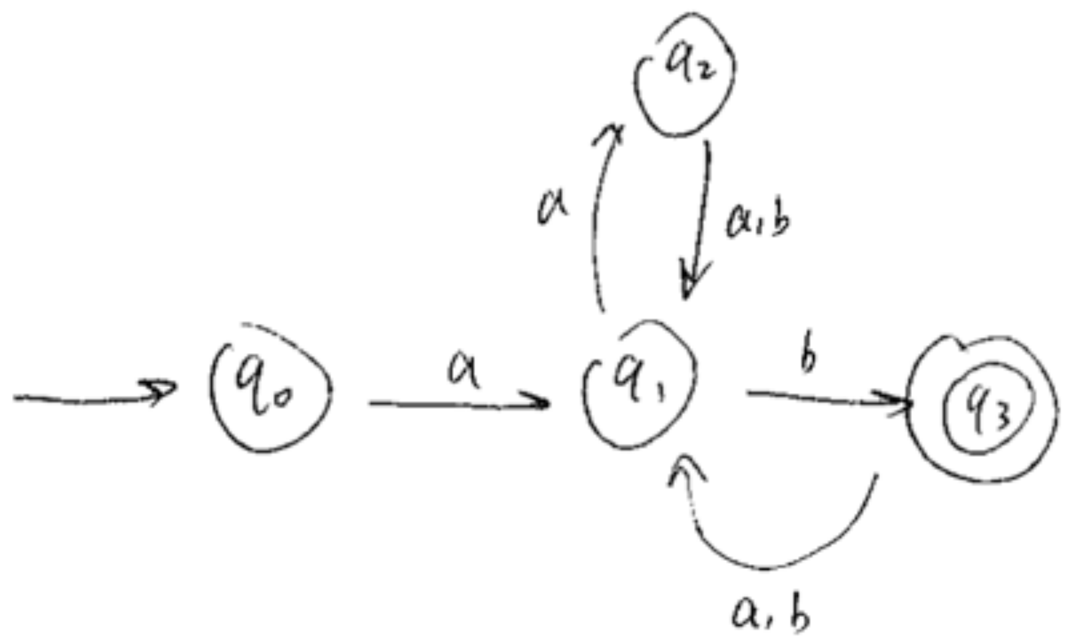
2. Prove that the language L = { awb | w \in {a,b}*, |w| is even } is regular.
(note that \in is the LaTeX symbol for "element of").

(2)

To prove $L = \{ awb \mid w \in \{a,b\}^*, |w| \text{ is even} \}$ is regular,

is to find some finite automaton recognizes it.

Consider machine M below. which recognizes L.



3. Prove that for every regular language L, its complement language is also regular.

Proof: For every regular language L, by the definition of regular language, there must be a DFA M s.t. $L(M) = L$. let $M = (Q, \Sigma, \delta, q, F)$ where δ is a **total** function, we construct another DFA $\overline{M} = (Q, \Sigma, \delta, q, \overline{F})$ where $\overline{F} = Q \setminus F$.

For every string $w \in L$, it will end up in a state $q \in F$ in M, and it will end up in the same state in $\overline{M}$ which rejects w since $q \notin \overline{F}$; similarly, for every string w' in the complement language, i.e., $w' \in \Sigma^* \setminus F$, it will end up in a state $q' \notin F$ in M, and it will end up in the same state in $\overline{M}$ which accepts w since $q' \in \overline{F}$. So $\overline{M}$ accepts all strings in $\Sigma^* \setminus L$ and **nothing else**, which means the complement language $\Sigma^* \setminus L$ is recognized by DFA $\overline{M}$, thus regular. $\square$

Note, however, that if δ is a partial function (i.e., trap state omitted), this proof does not work (why?). You would have to add a trap state and all trap transitions to make δ a total function first.