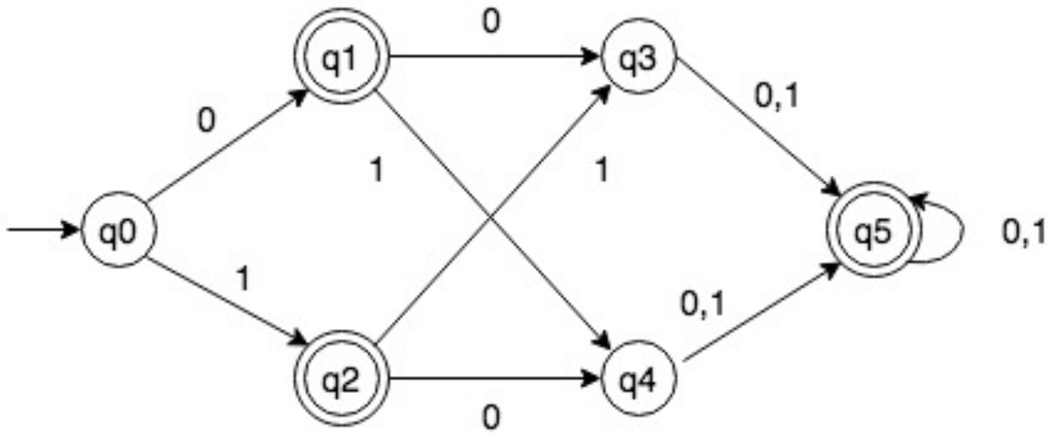
Solutions of Assignment #1 --- CSC320, Summer, 2018

Zhaocheng Li

V00832770

Q#1-(a):

recall the given graph,



With the instructions given in class, here we are going to trace the procedure to the solution, which is the minimal automaton.

The procedure is given as follow:

1. $\Pi \leftarrow \{F, Q - F\}$; refined $\leftarrow$ $true$;
2. **While** refined **do**
3. refined $\leftarrow$ $false$;
4. **For** $B \in \Pi$ **do**
5. **If** $\exists\, a$, $\exists\, p, q \in B$, $\exists\, B' \in \Pi$, **s.t.** $\delta(p, a) \in B'$ **and** $\delta(q, a) \notin B'$ **then**
6. $B_1 \leftarrow \{r \in B \mid \delta(r, a) \in B'\}$; $B_2 \leftarrow B - B_1$;
7. $\Pi \leftarrow (\Pi - \mathrm{B}) \cup B_1 \cup B_2$; refined $\leftarrow$ $true$;

Trace:

1. F = {q1, q2, q5}, Q-F = {q0, q3, q4}, a$\in${0, 1}
2. Π = {F, Q-F}, refined = true
   - refined = false
   - B = F, B'=Q-F
     a. a = 0, δ(q1, 0) = q3 $\in$ B'; δ(q2, 0) = q4 $\in$ B'; δ(q5, 0) = q5 $\notin$ B'
     b. B1 = {q1, q2}, B2 = {q5}
     c. Π = {{q1, q2}, {q5}, {q0, q3, q4}}, refined = true
   - B = Q-F, B' = {q5}
     a. a = 0, δ(q0, 0) = q1 $\notin$ B'; δ(q3, 0) = q5 $\in$ B'; δ(q4, 0) = q5 $\in$ B'

b. $B1 = \{q3, q4\}$, $B2 = \{q0\}$
c. $\Pi = \{\{q1, q2\}, \{q5\}, \{q3, q4\}, \{q0\}\}$, refined = true

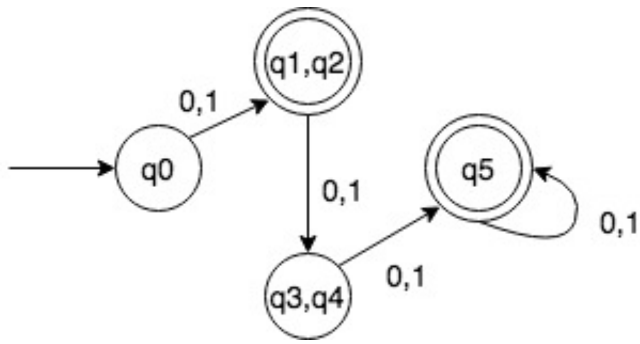
3. refined = false
4. PROGRAM TERMINATED

After the partitioning, we have $\Pi = \{\{q1, q2\}, \{q5\}, \{q3, q4\}, \{q0\}\}$ as final collection.

By constructing the minimal automaton $M' = (Q', \Sigma, \delta', [q0], F')$ where

- $Q' = \{\{q1, q2\}, \{q5\}, \{q3, q4\}, \{q0\}\}$
- $F' = \{\{q1, q2\}, \{q5\}\}$
- $\delta'([q], a) = [\delta(q, a)]$
- $[q0] = \{q0\}$

Hence, we have constructed such minimal DFA, visualizing as follow



Q#1-(b):

given the alphabet $\Sigma = \{0,1\}$, and the minimized DFA. Then the language accepted by this automaton is:

$\{w \mid w \in Q \text{ and } |w|=1 \text{ or } |w| > 2\}$

Q#2-(a)

Proof: given $L = \{w = 0^n1^m0^n \mid m, n \geq 0\}$, we give the proof by pumping lemma.

Take p as the "pumping length", and consider the string $w0=0^p1^q0^p$, such that $|w0| \geq p$, we can break down w0 into three strings, $w0=xyz$, satisfying the conditions:

1. $|xy| <= p$
2. $y \neq \varepsilon$
3. $xy^kz \in L$ for each $k \geq 0$

Now since $|xy| <= p$, it must be the case that $xy = 0^i$ for some $i \geq 0$. Since $y \neq \varepsilon$, it must be the case that $y=0^j$ with $j > 0$. Taking $k = 0$, so

$xy^kz = xy^0z = 0^{p-j}1^q0^p \notin L$, since there are more 0's behind 1's than 0's before 1's. Therefore, L is not regular.

Q#2-(b)

Proof: Given $L = \{0^m 1^n \mid m \neq n\}$. Here we can prove by contradiction.

Assuming that L is regular, and by the closure properties of regular languages. Let M be the complement of L and hence, M is regular.

Since $M = \{0^m 1^n \mid m=n\}$, consider the string $w = 0^p 1^p$, with pumping lemma p. Then $|w| = 2p > p$, we can break down w into three strings, w=xyz, satisfying the conditions:

1. $|xy| <= p$
2. $y \neq \varepsilon$
3. $xy^k z \in L$ for each $k \geq 0$

Now since $|xy| <= p$, it must be the case that $xy = 0^i$ for some $i \geq 0$. Since $y \neq \varepsilon$, it must be the case that $y=0^j$ with $j > 0$. Taking $k = 0$, so

$xy^k z = xy^0 z = 0^{p-j} 1^p \notin L$, the numbers of 0's and 1's are not equal, contradicting the assumption that M is regular.

So, L is not regular.

Q#2-(c)

Proof: Given $L = \{wtw \mid w, t \in \{0, 1\}^+\}$

Consider $wtw = 0^p 1 0^p$, with pumping lemma p. such that $|wtw| \geq p$, we can break down wtw into three strings, wtw=xyz, satisfying the conditions:

1. $|xy| <= p$
2. $y \neq \varepsilon$
3. $xy^k z \in L$ for each $k \geq 0$

Now since $|xy| <= p$, it must be the case that $xy = 0^i$ for some $i \geq 0$. Since $y \neq \varepsilon$, it must be the case that $y=0^j$ with $j > 0$. Taking $k = 0$, so

$xy^k z = xy^0 z = 0^{p-j} 1 0^p \notin L$, since there are more 0's behind 1's than 0's before 1's. So, L is not regular.

Q#3-(a):

given the language $L = \{w \in \{0, 1\}^* \mid w = w^R\}$, the CFG could be the set $G = (\{S\}, \{0, 1\}, P, S)$, and the set of P of productions would be:

- $S \rightarrow \varepsilon$
- $S \rightarrow 0$
- $S \rightarrow 1$
- $S \rightarrow 0S0$
- $S \rightarrow 1S1$

Q#3-(b):

given the language $L = \{w \in \{0, 1\}^* \mid$ w contains the same number of 0's and 1's$\}$, the CFG could be the set $G = (\{S\}, \{0, 1\}, P, S)$, and the set of P of productions would be:

- $S \rightarrow \varepsilon$

- $S \to 1S0$
- $S \to 0S1$

Q#3-(c):

given the language $L = \{w \in \{0, 1\}^* \mid w = 0^n1^n, n \geq 0\}$, the CFG could be the set $G = (\{S\}, \{0, 1\}, P, S)$, and the set of P of productions would be:

- $S \to \varepsilon$
- $S \to 0S1$

Q#4:

Given the language $A = \{a^ib^jc^k \mid i = j \text{ or } j = k \text{ where } i, j, k \geq 0\}$. We have the corresponding CFG G=({A, B, C, D, E}, {a, b, c}, P, S), and the set of P of productions would be:

- $A \to B$
- $A \to C$
- $B \to aB$
- $B \to D$
- $C \to Cc$
- $C \to E$
- $D \to \varepsilon$
- $D \to bDc$
- $E \to \varepsilon$
- $E \to aEb$

This CFG is ambiguous because we can find two distinct leftmost derivations, one is

- $A \Rightarrow C \Rightarrow Cc \Rightarrow aEbc \Rightarrow abc$

And the other could be

- $A \Rightarrow B \Rightarrow aB \Rightarrow abBc \Rightarrow abc$

Q#5:

Given the grammar;

- $E \to E + T \mid T$
- $T \to T * F \mid F$
- $F \to (E) \mid num$

Into the CNF form:

- $S0 \to E$
- $E \to EA$
- $A \to RB$
- $B \to RT$
- $T \to TC$
- $C \to WD$

- $D \to WF$
- $F \to num \mid YP \mid YZ$
- $P \to EQ$
- $Q \to EZ$
- $R \to +$
- $W \to *$
- $Y \to ($
- $Z \to)$

Q#6:

Given the CNF form for the grammar

$$E \to E * E \mid E + E \mid (E) \mid id \mid num$$

as:

- $E \to EA \mid EB \mid LD \mid id \mid num$
- $A \to ME$
- $B \to PE$
- $D \to ER$
- $M \to *$
- $P \to +$
- $L \to ($
- $R \to)$

Consider CYK algorithm on w = (id + num) * num

|w| = 7

We have a table T that is 7X7 (UPPER TRIANGLE ONLY) and V = { S0, E, A, B, D, M, P, L, R}, if we have s0 as the start variable

L	LE	LEP	LEP	LPER	LPERM	LPERM
	E	EP	EP	PER	PERM	PERM
		P	PE	PER	PERM	PERM
			E	ER	ERM	RME
				R	RM	RME
					M	ME
						E

Q#7: ALL CNF form are unambiguous, since

1. Unique new start variable
2. There is no more null terminal, so the order of each terminal in the fixed direction of derivation is fixed.
3. Each variable is assigned either one terminal or two variables, along each way you either reach to the end or further without loop back and termination.