Q1)

a. Enumerating positive & negative integers:

	1	2	3	4	5	6...
-1	x					
-2	$-1/2$	x				
-3	$-1/3$	$-2/3$	x			
-4	$-1/4$	$-2/4$	$-3/4$	x		
-5	$-1/5$	$-2/5$	$-3/5$	$-4/5$	x	
-6	$-1/6$	$-2/6$	$-3/6$	$-4/6$	$5/6$	...
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

devising a mapping between $\mathbb{N}^+$ and $\mathbb{Q}' = \{ a/b : 0 > a/b > -1, \ a,b \in \mathbb{Z} - \{0\} \}$:

$1 \leftrightarrow -1/2$ } $\sum abs(a,b) = 3$

$2 \leftrightarrow -1/3$ } $\sum abs(a,b) = 4$

$3 \leftrightarrow -1/4$ } $\sum abs(a,b) = 5$

$4 \leftrightarrow -2/3$ } $\sum abs(a,b) = 5$

$5 \leftrightarrow -2/4$, $6 \leftrightarrow -1/5$ } $\sum abs(a,b) = 6$

$7 \leftrightarrow -1/6$, $8 \leftrightarrow -2/5$, $9 \leftrightarrow -3/4$ } $\sum abs(a,b) = 7$

...

This mapping is 1-to-1 and onto as each group satisfying the condition $\sum(abs(a), abs(b))$ comprises of finitely-many elements and an infinite union of finitely many elements results in a <u>countably-infinite</u> set!

b. All finite languages are regular. Consequently, L^+ in question constitutes a regular language, and it can be represented by some regular expression. Thus, portion of L^+ that cannot be represented by reg. expr. is the empty set, which is <u>finite</u> with <u>cardinality zero</u>.

c. The set of all languages, $P(N)$ is a well-known uncountable set. The set of languages that can be recognized by FA is a countable set as there are countably-many infinite FA's, where each FA can be encoded as finite strings based on some encoding scheme.

utilizing the formal five-tuple representation of the FA in question.

Let C be the set of all languages and
R be the set of regular languages over $\{a, b\}$.

We claim that C is uncountable and R is countable, and we are asked about $C - R$.

Assume that $C - R$ is countable.
Then the set $(C-R) \cup R$ should also be countable due to being union of two countable sets. Via set theory and our knowledge of existence of the set of languages that are not regular,

$$C \subseteq (C-R) \cup R,$$

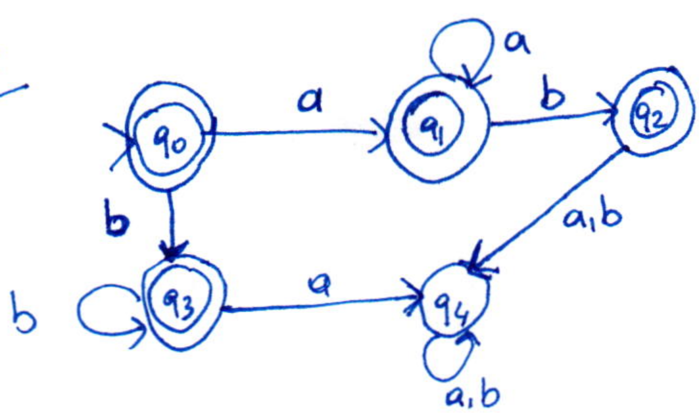
then since $(C-R) \cup R$ is shown to be countable due to our assumption about $C-R$, and C is contained within $(C-R) \cup R$, C should also be countable.
However, C is absolutely uncountable, and we have to refute our initial assumption that $C-R$ is countable.
Consequently, $C-R$ is <u>uncantable.</u>

Q.2) a. $M_1 = (K_1, \{a, b\}, \delta_1, q_0, F_1)$ where

$K_1 = \{q_0, q_1, q_2, q_3, q_4\}$,

$F_1 = \{q_0, q_1, q_2, q_3\}$.

$\underline{\delta_1:}$



claiming $L(M_1) = L$.
(no need to formally show)

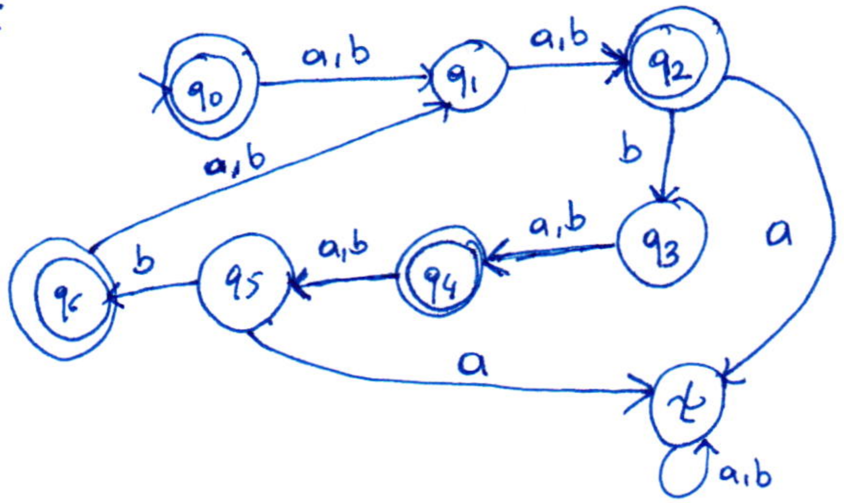
b. $M_2 = (K_2, \{a, b\}, \delta_2, q_0, F_2)$ where

$K_2 = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, t\}$

$F_2 = \{q_0, q_2, q_4, q_6\}$

and

δ_2 :



and $L(M_2) = L_2$.

c. $M_3 = (K_3, \{a, b, c\}, \delta_3, q_0, F_3)$ where

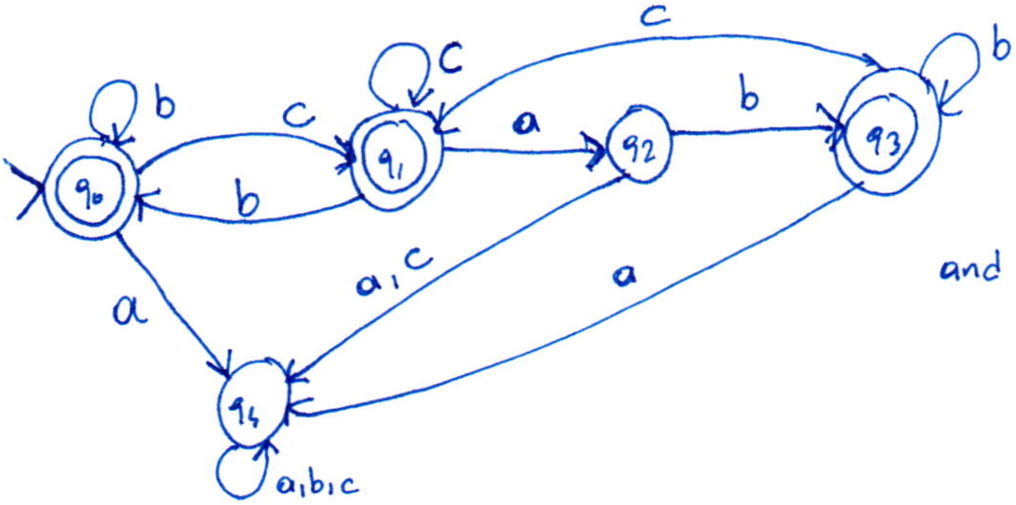
$K_3 = \{q_0, q_1, q_2, q_3, q_4\}$

$F_3 = \{q_0, q_1, q_3\}$

in fact, q_3 can be eliminated.

and

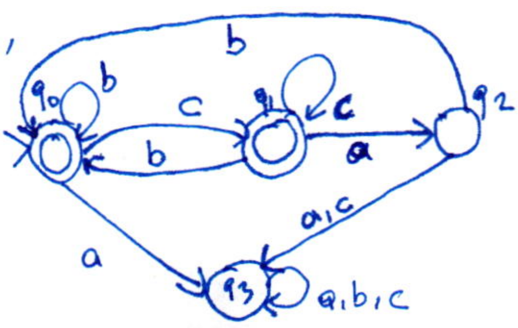
δ_3 :



and $L(M_3) = L_3$.

alternatively,

δ_3' :

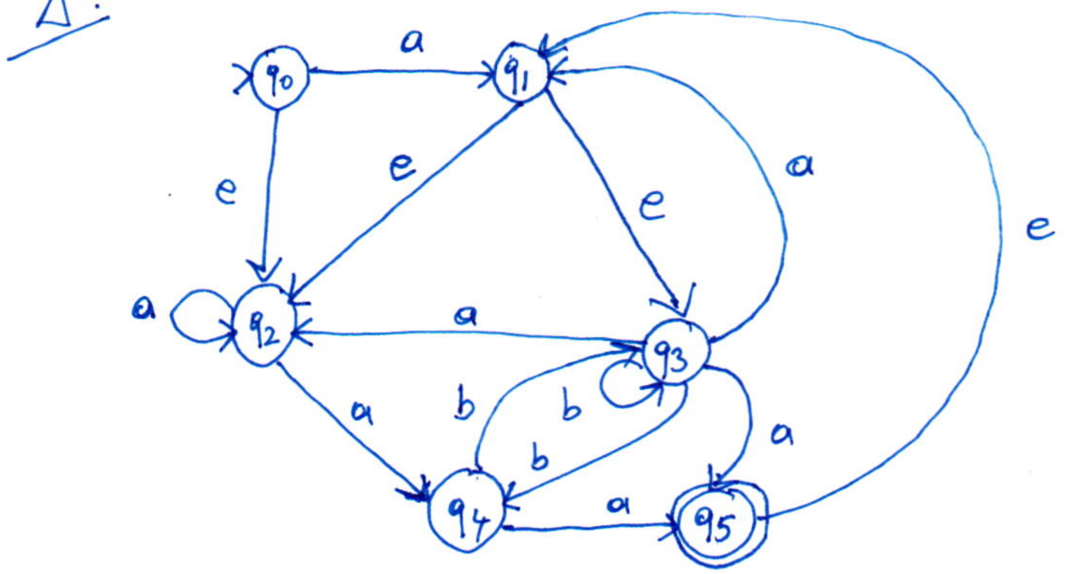


$M_3' = (\{q_0, q_1, q_2, q_3\}, \{a, b, c\}, \delta_3', q_0, \{q_0, q_1\})$

and $L(M_3') = L_3$.

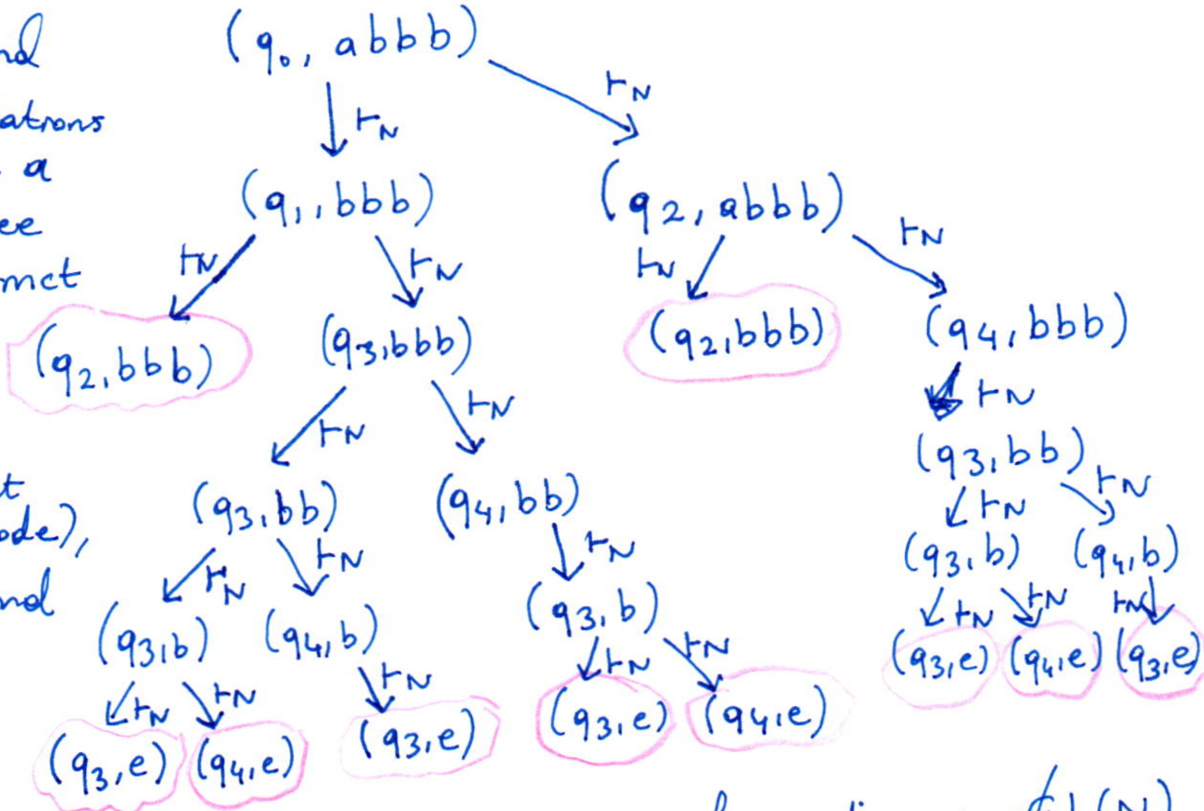
Q3)

Δ :



a.) $w_1 = abbb$.

Configuration and entailment relations are shown in a computation tree to indicate distinct choices made by N, when it reaches an end (some branch that reaches a leaf node), it backtracks and tries an alternative.



Since no branch reaches an accepting configuration, $w_1 \notin L(N)$.

b.) $w_2 = ababa$.

$(q_0, ababa) \vdash_N (q_1, baba) \vdash_N (q_3, baba) \vdash_N (q_3, aba) \vdash_N$

$(q_1, ba) \vdash_N (q_3, ba) \vdash_N (q_3, a) \vdash_N (q_5, e)$.

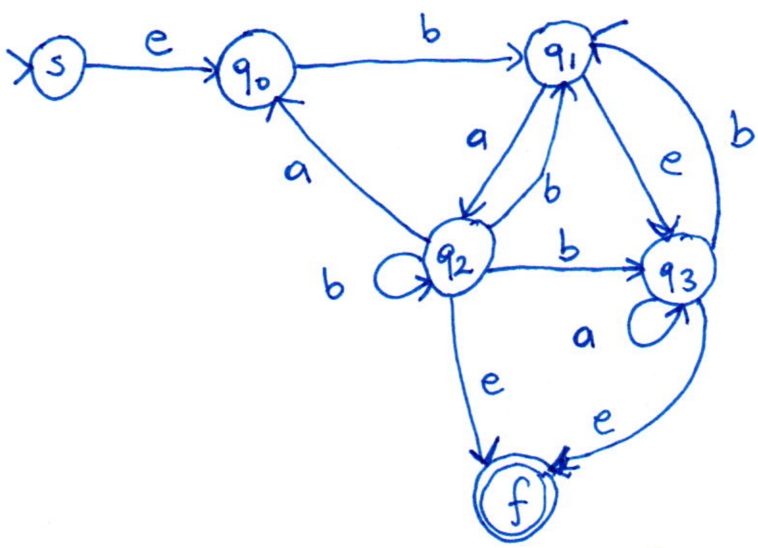
Since $q_5 \in F$, $w_2 \in L(N)$.

Q.4)

a. $G_N = (K_N, \Sigma, \Delta_N, s, F_N)$ where

$K_N = K \cup \{s, f\}$, $\Delta_N = \Delta \cup \{(s, e, q_0), (q_2, e, f), (q_3, e, f)\}$,

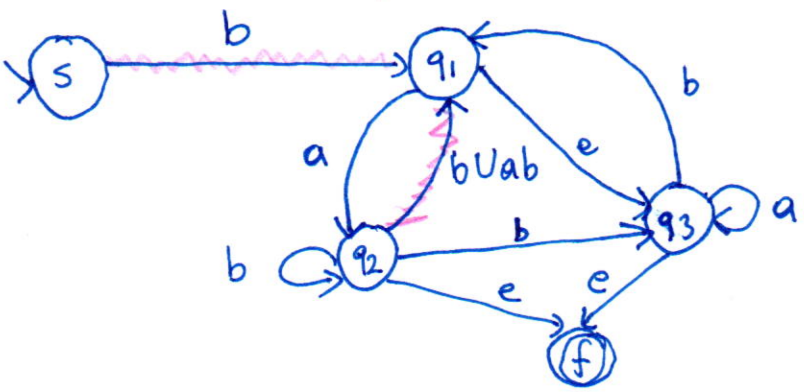
and $F_N = \{f\}$ and graphical representation is included below:



b. eliminating states of G_N one by one except for $\{s, f\}$.

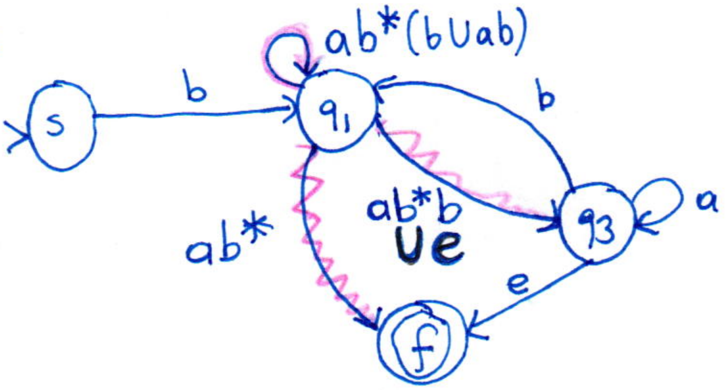
eliminate q_0: (and all incoming outgoing edges)	incoming s, q_2	outgoing q_1 (except for self-loops)	$\Rightarrow$	work on paths (s, q_0, q_1) and (q_2, q_0, q_1).

G_N:



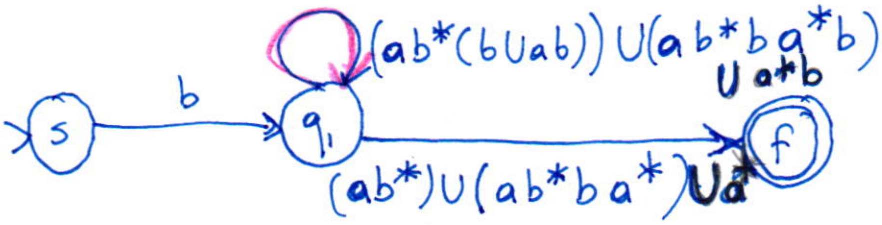
eliminate q_2:	incoming q_1	outgoing q_1, q_3, f	$\Rightarrow$	paths to be updated: (q_1, q_2, q_1) (q_1, q_2, q_3) (q_1, q_2, f)

G_N:



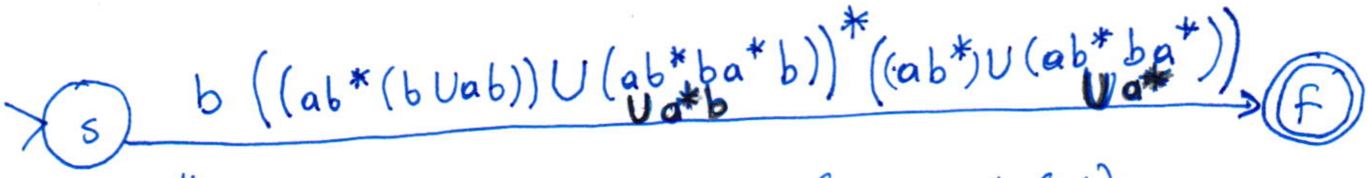
<u>eliminate q_3:</u> incoming q_1 outgoing q_1, f $\Rightarrow$ working on: (q_1, q_3, q_1) (q_1, q_3, f)

G_N:



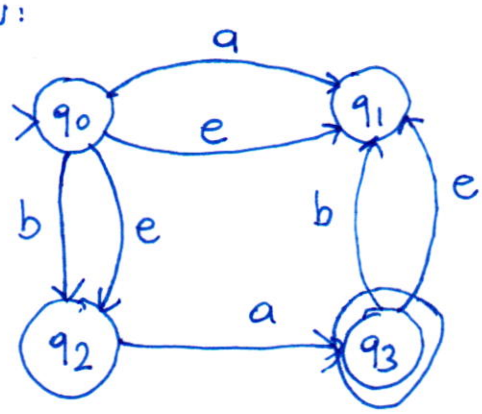
<u>eliminate q_1:</u>

G_N:



$\Downarrow$ is the regular expression equivalent to $L(N)$.

Q.5) N:



a.) $E(q_0) = \boxed{\{q_0, q_1, q_2\}}$.

$\delta(\{q_0, q_1, q_2\}, a) = E(q_1) \cup E(q_3) = \boxed{\{q_1, q_3\}}$.

$\delta(\{q_0, q_1, q_2\}, b) = E(q_2) = \boxed{\{q_2\}}$.

$\delta(\{q_1, q_3\}, a) = \boxed{\emptyset}$.

$\delta(\{q_1, q_3\}, b) = E(q_1) = \boxed{\{q_1\}}$.

$\delta(\{q_2\}, a) = E(q_3) = \{q_1, q_3\}$.

$\delta(\{q_2\}, b) = \emptyset$.

$\delta(\{q_1\}, a) = \emptyset$.

$\delta(\{q_1\}, b) = \emptyset$.

$\delta(\emptyset, a) = \emptyset$.

$\delta(\emptyset, b) = \emptyset$.

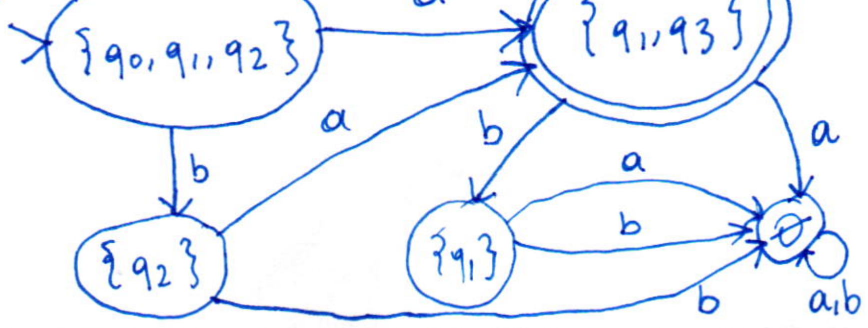
M is the DFA output of the subset construction algorithm

$\Uparrow$

Let $M = (K_M, \{a,b\}, \delta, \{q_0, q_1, q_2\}, \{\{q_1, q_3\}\})$

with $K_M = \{\{q_0, q_1, q_2\}, \{q_1, q_3\}, \{q_2\}, \{q_1\}, \emptyset\}$, $L(M) = L(N)$.

M:

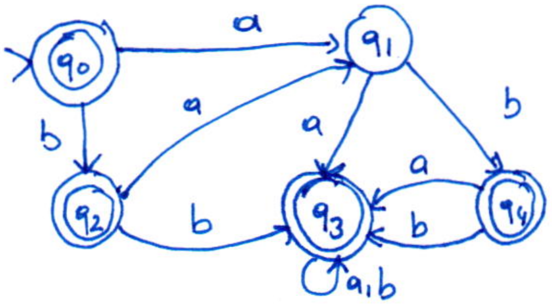


6. Regular languages are closed under complementation.

A DFA to recognize $L = \Sigma^* - L(N)$ can be constructed by switching final and nonfinal states of M as follows:

M_R :

δ_R:



$M_R = (\{q_0, q_1, q_2, q_3, q_4\}, \{a, b\}, \delta_R, q_0, \underbrace{\{q_0, q_2, q_3, q_4\}}_{:= F_R})$

and $L(M_R) = \overline{L(N)}$.

You can convert this DFA to some reg. expr. w/o explicitly showing the steps. I will do as follows for educational purposes:

1 - Convert M_R into a DFA with minimal # of states.

2 - Do the GNFA $\rightarrow$ reg. expr. conversion

$\equiv_0$: $\{q_0, q_2, q_3, q_4\}$, $\{q_1\}$ // q_1 final, the rest nonfinal

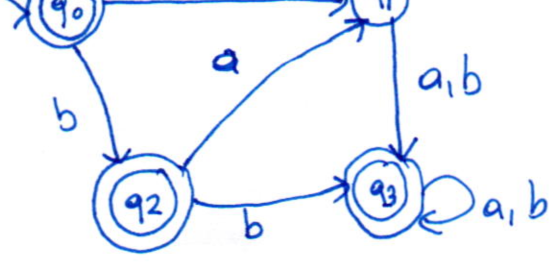
$\equiv_1$: $\{q_0, q_2\}$, $\{q_3, q_4\}$, $\{q_1\}$ // $\delta'_R(q_0, a) \in \{q_1\}$, $\delta'_R(q_0, b) \in F_R$
// $\delta'_R(q_2, a) \in \{q_1\}$, $\delta'_R(q_2, b) \in F_R$
// but
// $\delta'_R(\{q_3, q_4\}, a) \in F_R$ and
// $\delta'_R(\{q_3, q_4\}, b) \in F_R$.

$\equiv_2$: $\{q_0\}$, $\{q_2\}$, $\{q_3, q_4\}$, $\{q_1\}$ // $\delta'_R(\{q_0\}, a) \in \{q_1\}$
// $\delta'_R(\{q_2\}, a) \in \{q_1\}$ BUT
// $\delta'_R(\{q_0\}, b) \in \{q_0, q_2\}$, yet
// $\delta'_R(\{q_2\}, b) \in \{q_3, q_4\}$.

no further refinements are possible; so

M'_R :

δ'_R:



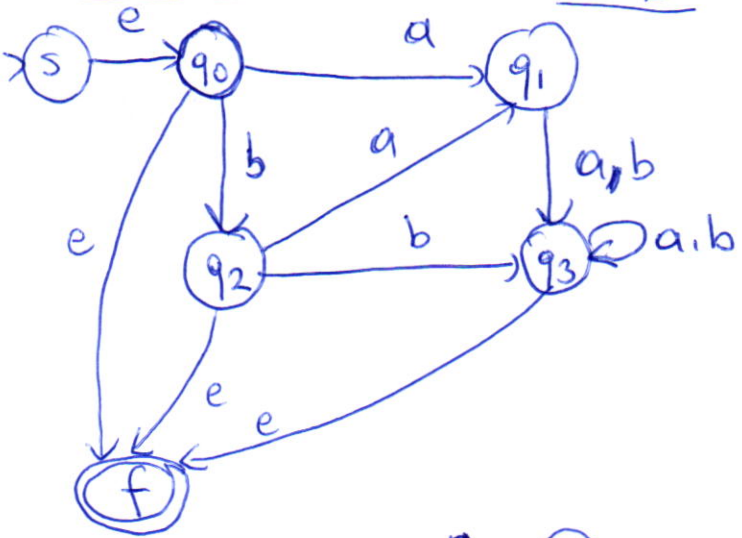
with

$M'_R = (\{q_0, q_1, q_2, q_3\}, \{a, b\}, \delta'_R, q_0, \{q_0, q_2, q_3\})$

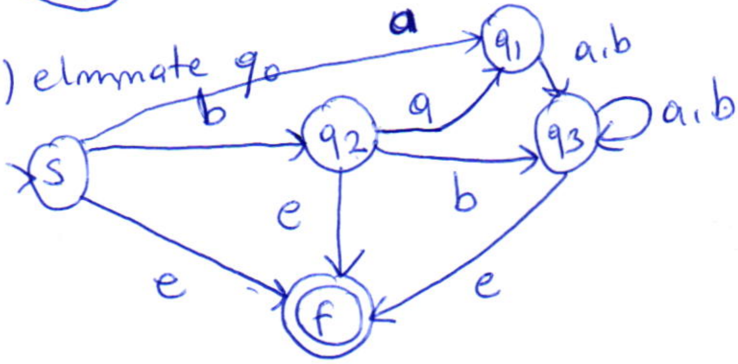
is the DFA with minimal number of states to recognize $L(M_R)$.

GNFA for M'_R:

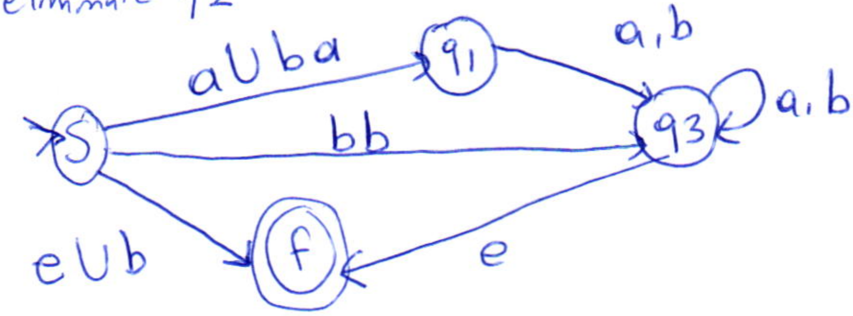
$G_{M'_R}$:



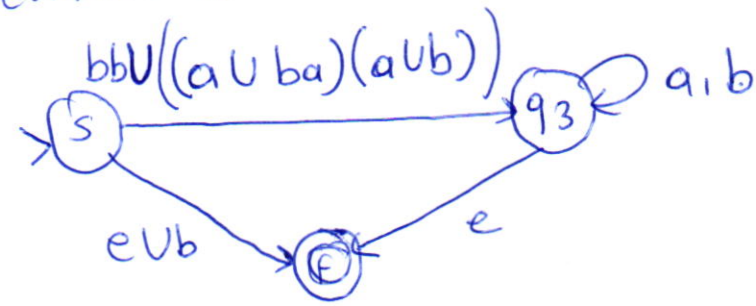
i) eliminate q_0



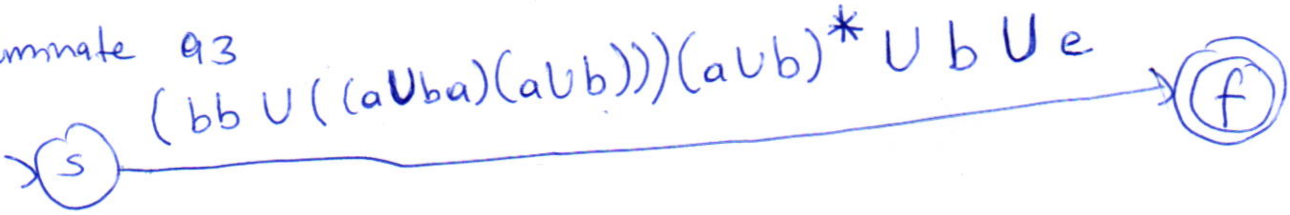
ii) eliminate q_2



iii) eliminate q_1



iv) eliminate q_3



Translate RE into <u>set</u>:

$\mathcal{L}(\emptyset \cup b) = \{e, b\}$

$\mathcal{L}((a \cup b)^*) = \{a, b\}^*$

$\mathcal{L}\left(bb \cup ((a \cup ba)(a \cup b))\right) = \{aa, ab, bb, baa, bab\}$

$\mathcal{L}\left((bb \cup ((a \cup ba)(a \cup b)))(a \cup b)^*\right) =$

$\{aa, ab, bb, baa, bab\} \cdot \left(\bigcup_{i=0}^{\infty} \{a, b\}^i\right)$

$= \{aa, ab, bb, baa, bab\} \cup \{aa, ab, bb, baa, bab\}\{a, b\}$

$\cup \{aa, ab, bb, baa, bab\}\{a, b\}\{a, b\} \cup \ldots$

$\mathcal{L}\left((bb \cup ((a \cup ba)(a \cup b)))(a \cup b)^* \cup b \cup e\right) =$

$\{e, b\} \cup \{aa, ab, bb, baa, bab\} \cup \left(\{aa, ab, bb\} \cup \{baa, bab\}\right)\{a, b\}$

$\cup \left(\{aa, ab, bb\} \cup \{baa, bab\}\right)\{a, b\}\{a, b\} \cup \ldots$

$= \{e, b, aa, ab, bb, baa, bab\} \cup \{aa, ab, bb, ba\}\{a, b\}\{a, b\}^*$

$= \{e, b, aa, ab, bb, baa, bab\} \cup \{aaa, aab, aba, abb, baa, bab, \underbrace{bba, bbb}_{\{a,b\}\{a,b\}\{a,b\}}\}\{a, b\}^*$

$= \left(\{e\} \cup \{a, b\} \cup \{a, b\}\{a, b\} \cup \{a, b\}\{a, b\}\{a, b\} \cup \{a, b\}\{a, b\}\{a, b\}\{a, b\} \cup \ldots\right) - \{a, ba\}$

$= \{a, b\}^* - \{a, ba\}$

$\Rightarrow L_{M_R}' = L_{M_R} = \{w \in \{a, b\}^* : w \neq a \text{ and } w \neq ba\}.$

Q.6) Assume that there exists NFA $M_1 = (K_1, \Sigma, \Delta_1, s_1, F_1)$ and NFA $M_2 = (K_2, \Sigma, \Delta_2, s_2, F_2)$ s.t. $L(M_1) = L_1$ and $L(M_2) = L_2$. (Since L_1 and L_2 are regular languages, some NFA to recognize them must exist.).

By set theorem

$$L_1 - L_2 \equiv L_1 \cap \overline{L_2}$$
$$\equiv \overline{\left(\overline{L_1} \cup \overline{\overline{L_2}}\right)} \quad \text{(by De Morgan laws)}$$
$$\equiv \overline{(\overline{L_1} \cup L_2)}.$$

There exists a DFA $M_1^D = (K_1^D, \Sigma, \delta_1^D, s_1^D, F_1^D)$ to recognize L_1 via the equivalence of DFA and NFA in which M_1^D can be constructed by the subset construction algorithm taking M_1 as its input, ensuring that $L(M_1^D) = L(M_1) = L_1$.

Then, we can construct $M_1'^D = (K_1^D, \Sigma, \delta_1^D, s_1^D, K_1^D - F_1^D)$ by switching final and nonfinal states of M_1^D. By the theorem of regular languages being closed under complementation $L(M_1'^D) = \Sigma^* - L(M_1^D) = \Sigma^* - L_1 = \overline{L_1}$.

As regular languages are closed under union, we can construct an NFA $M_{12} = (K_1^D \cup K_2 \cup \{s\}, \Sigma,$
$\Delta_2 \cup \delta_1^D \cup \{(s, e, s_1^D), (s, e, s_2)\},$
$s, (K_1^D - F_1^D) \cup F_2)$

s.t. $L(M_{12}) = \overline{L_1} \cup L_2$

as established by the relevant theorem.

Then, again we carry-out NFA-to-DFA conversion between M_{12} and M_{12}^D, the equivalent DFA s.t.

$M_{12}^D = (K_{12}^D, \Sigma, \delta_{12}^D, s_{12}^D, F_{12}^D)$ with $L(M_{12}^D) = L(M_{12})$.

Finally we switch final and nonfinal states of M_{12}^D to construct the DFA $M_{12}'^D = (K_{12}^D, \Sigma, \delta_{12}^D, s_{12}^D, K_{12}^D - F_{12}^D)$ s.t.

$$L(M_{12}'^D) = \Sigma^* - L(M_{12}^D) = \Sigma^* - (\overline{L_1} \cup L_2)$$
$$= \overline{(\overline{L_1} \cup L_2)} = L_1 - L_2$$

as ensured by the theorem proving that regular languages are closed under complementation.

For any NFA M_1, M_2 s.t. $L(M_1) = L_1$ and $L(M_2) = L_2$, we can carry out this construction procedure to arrive at some DFA (which is trivially an NFA) to recognize $L_1 - L_2$ ∎

Q.7)

a. Assume that L is regular. (Note that L is an infinite language.)

Take $w = a^{n^2} \in L$ for some $n \geq 0$.

$|w| = n^2 \geq n$.

Decompose $w = xyz$, s.t. $y \neq e$. Via pumping lemma $|xy| \leq n$ and $|y| \geq 1$ should hold.

Consequently, $w = a^{n^2} = a^i a^j a^k$, $i+j+k = n^2$ and $j \geq 1$ and $i+j \leq n$, $n^2-n \leq k \leq n^2-1$

and $\begin{cases} x = a^i, \\ y = a^j, \\ z = a^k. \end{cases}$

P.L. for regular languages states that for $t = 0, 1, 2, \ldots$ $xy^tz \in L$;

$xy^tz = a^i a^{jt} a^k$, and

if $t = 0 \Rightarrow xz = a^i a^k = a^{i+k} \in L$ should hold.

Note that $n^2-n \leq k \leq n^2-1$ and $0 \leq i \leq n-1$

implying that $n^2-n \leq k+i \leq n^2+n-2$

Note $\boxed{(\cancel{n-1})^2 < n^2-n \leq k+i \leq n^2+n-2 < (n+1)^2}$

and since $j \geq 1$, $\boxed{k+i \neq n^2}$. Consequently, for every decomp. $a^{i+k} \notin L$, and we should refute our assumption that L is regular. L is not regular. ∎

Q7) b) $L = \{ w \in \{a,b\}^* \cdot f(a,w) = n^2$ for some $n \in \mathbb{N} \}$.

Define $L' = L \cap \{a^*\} = \{ a^{n^2} : n \in \mathbb{N} \}$

Assume that L is regular.

Then, since RL's are closed under intersection, L' is also regular.

~~Myhill~~-Nerode theorem states that being a regular language, L' must be the union of some of the equivalence classes of a right-invariant equivalence relation of finite index; since some DFA with minimal number of states to ~~recognize~~ L' exists.

Finiteness of the equivalence relation suggests that

$$a^k \approx_{L'} a^p \quad \text{for some } k, p \in \mathbb{N} \text{ s.t. } p > k.$$

Then, $t = p - k > 0$.

For any $i \geqslant t$, and $i^2 \geqslant k$, $i \in \mathbb{N}$;

via right-invariance we get

$$\underbrace{a^k a^{i^2 - k}}_{a^{i^2} \in L'} \approx_{L'} \underbrace{a^p a^{i^2 - k}}_{a^{i^2 + p - k}} = a^{i^2 + t} \notin L'$$

since $\boxed{i^2 < i^2 + t < i^2 + 2i + 1 = (i+1)^2}$

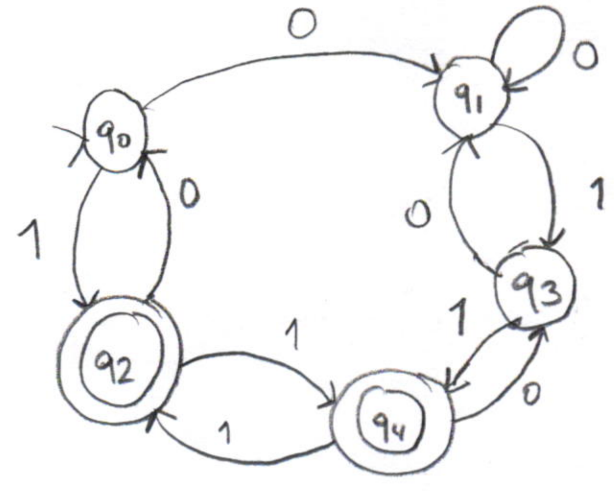
as $t \leq i < 2i + 1$.

Since there are infinitely many a^k, a^p with $p > k$, there exists infinitely many equivalence classes of L'.

Therefore L' cannot be regular, implying that L cannot be regular as well ∎

Q8)

M:



a. $\equiv_0$: $\{q_0, q_1, q_3\}, \{q_2, q_4\}$ as $q_0, q_1, q_3 \in K-F$ and $q_2, q_4 \in F$.

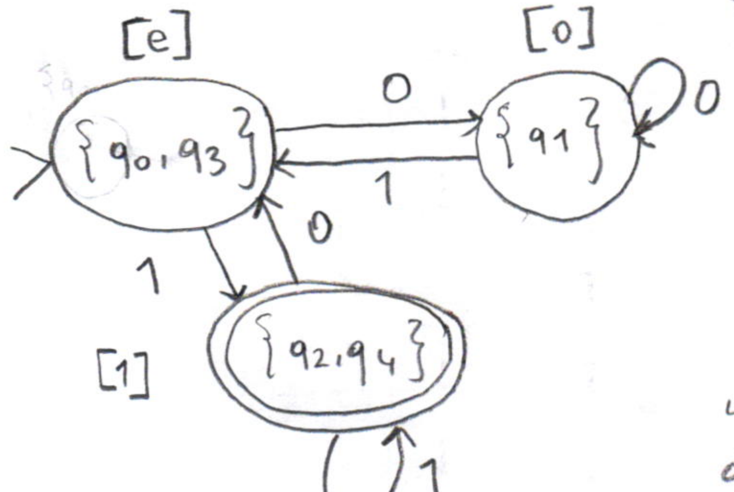
$\equiv_1$: $\{q_0, q_3\}, \{q_1\}, \{q_2, q_4\}$ as $\delta(q_0, 0) \in \{q_0, q_1, q_3\}$
$\delta(q_1, 0) \in \{q_0, q_1, q_3\}$
$\delta(q_3, 0) \in \{q_0, q_1, q_3\}$
BUT
$\delta(q_0, 1) \in \{q_2, q_4\}$
$\delta(q_1, 1) \in \{q_0, q_1, q_3\} \Rightarrow$ new group
$\delta(q_3, 1) \in \{q_2, q_4\}$

$\equiv_2$: $\{q_0, q_3\}, \{q_1\}, \{q_2, q_4\}$

no more refinements are possible...

⇓

M' :



$M' = (\{\{q_0, q_3\}, \{q_1\}, \{q_2, q_4\}\}, \{0, 1\}, \delta', \{q_0, q_3\}, \{\{q_2, q_4\}\})$

where δ' is graphically depicted is the DFA with minimal # of states equivalent to M.

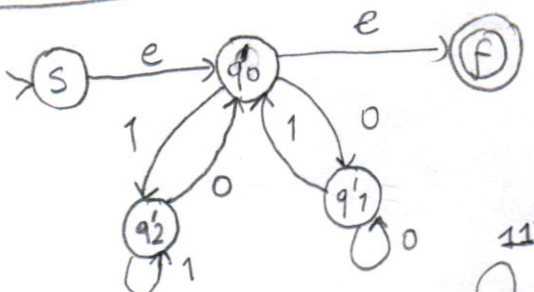
b. Three states of M' partition Σ^* into three equivalence classes that are written on the graph depicting M'.

$$\Sigma^* = [e]_{M'} \cup [0]_{M'} \cup [1]_{M'}$$

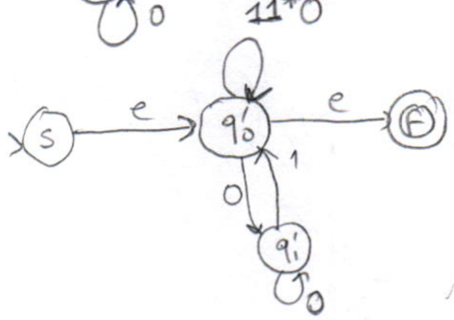
We need to write regular expressions for each equivalence class.

A mechanical procedure involves
→ creating a GNFA for each state of M' made an accepting state subsequently,
→ converting all GNFA into reg exp.

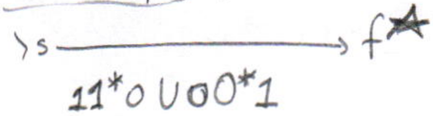
Computing [e] :



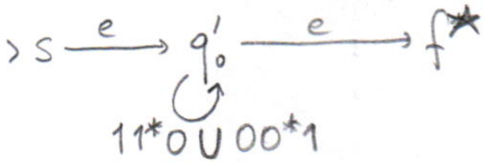
eliminate q'2 :



eliminate q'0 :



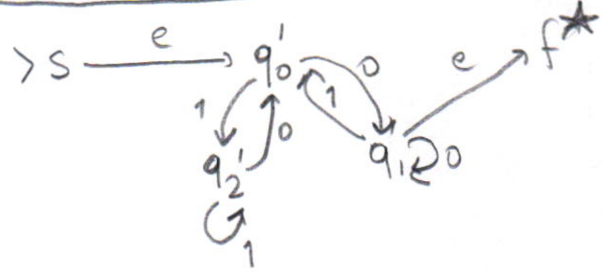
eliminate q'1



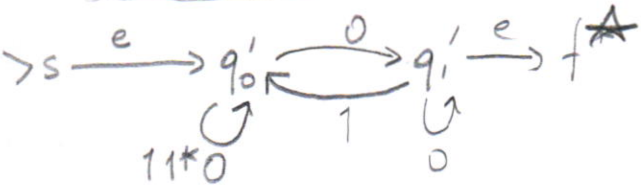
Thus,

$[e] = 11^*0 \cup 00^*1$.

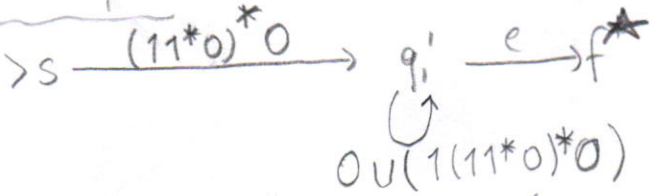
Computing [0] :



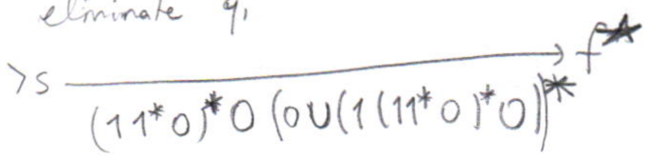
eliminate q'2 :



eliminate q'0 :

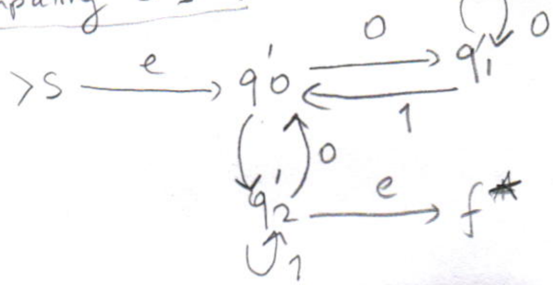


eliminate q1'

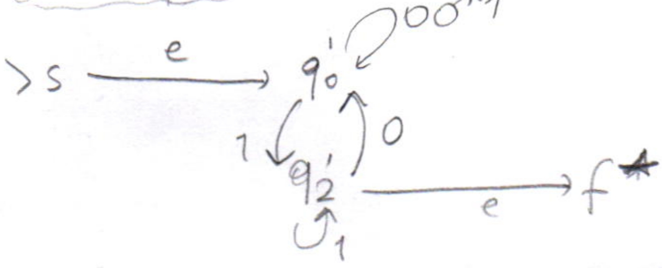


$[0] = (11^*0)^*0\,(0 \cup (1\,(11^*0)^*0))^*$.

Computing [1] :



eliminate q'1



eliminate q'_0 :

$$\begin{tikzcd} >s \arrow[r, "(00^*1)^*1"] & q'_2 \arrow[loop below, "0(00^*1)^*1 \cup 1"] \arrow[r, "e"] & f^* \end{tikzcd}$$

eliminate q'_2 :

$$\begin{tikzcd} >s \arrow[r, "(00^*1)^*1\,[0(00^*1)^*1 \cup 1]^*"'] & f^* \end{tikzcd}$$

Thus $[1] = (00^*1)^*1\left(0(00^*1)^*1 \cup 1\right)^* = L = L(M) = L(M')$.

and $L' = [e] \cup [1]$

$= (11^*0) \cup (00^*1) \cup \left((11^*0)^*0\,(1(11^*0)^*00 \cup 0)^*\right)$.

and $L \cup L' = \Sigma^*$.