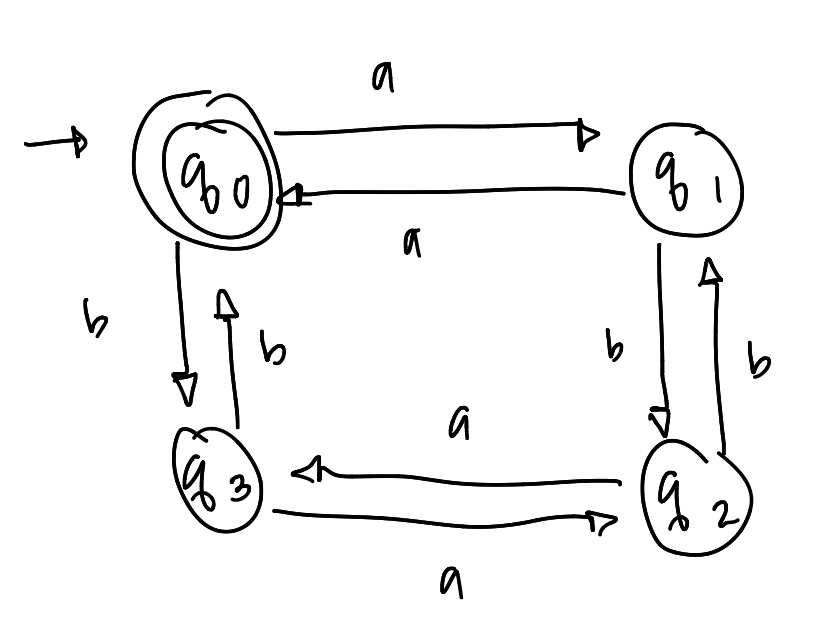
CSci 435: Formal Languages and Automata

Instructor: Dr. M. E. Kim Name: Elena Corpus

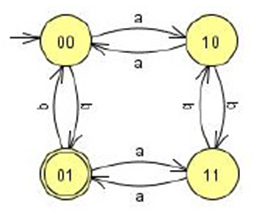
**Home Assignment 1: 76/110 points + 10 points (optional)**

Q1. [15/25] For S = {a, b}, construct the minimal DFA that accept the language consisting of

1. **[6/8**] all strings with an even number of *a*’s and an odd number of *b*’s.

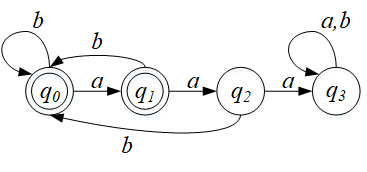


* This system does not accept all strings with an even number of *a*’s and an odd number of *b*’s.
* q3 is a final state, not a q0.

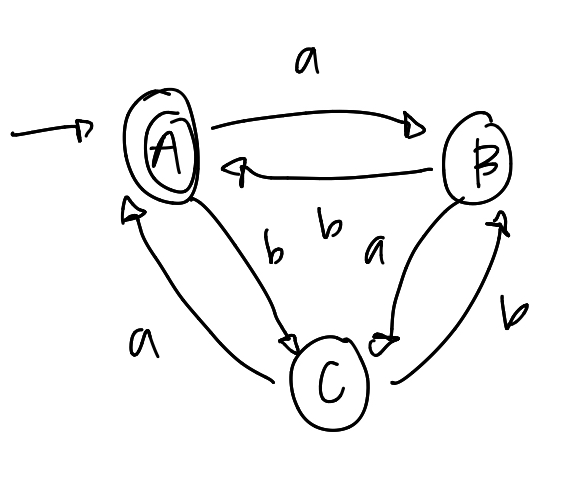


1. [0/8] every ‘*aa’* is followed immediately by a ‘*b’*. For example, the strings *aab*, *aaba*, *aabaabbaab* are in the language, but *aaab* and *aabaa* are not. Construct a DFA with 4 states.

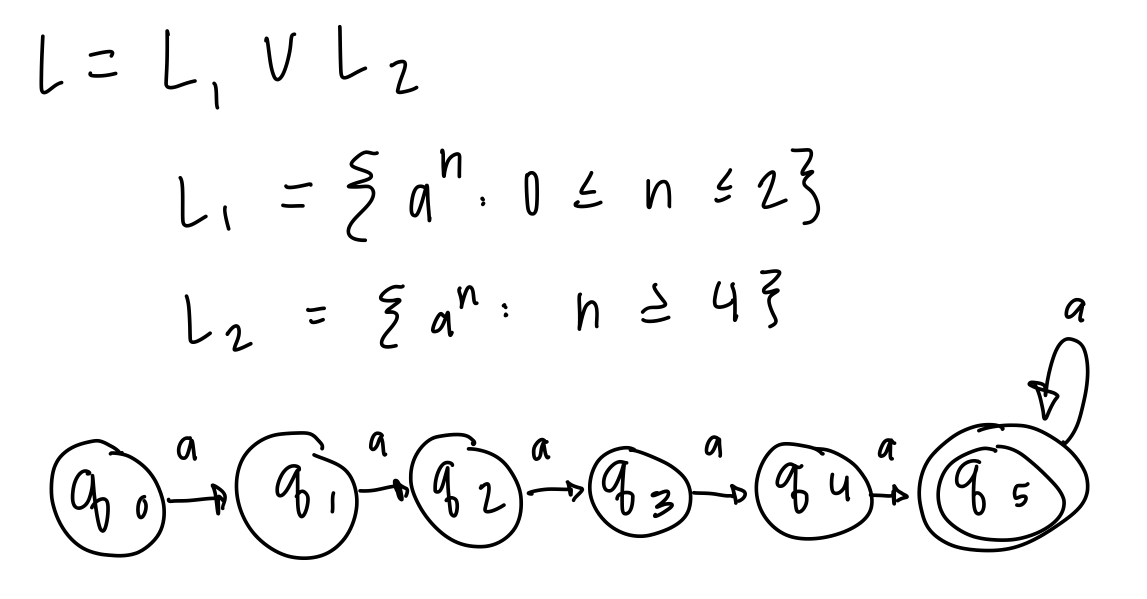
* See the attached solution



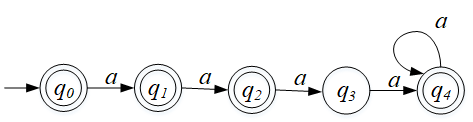
1. [9/9] L = {w | ( *na*(*w*) – *nb*(*w*) ) mod 3 = 0 }. Construct a DFA with 3 states.



Q2. [5/10] Show that the language L = { *a****n***| *n* ³ 0, *n* ¹ 3 } is regular.

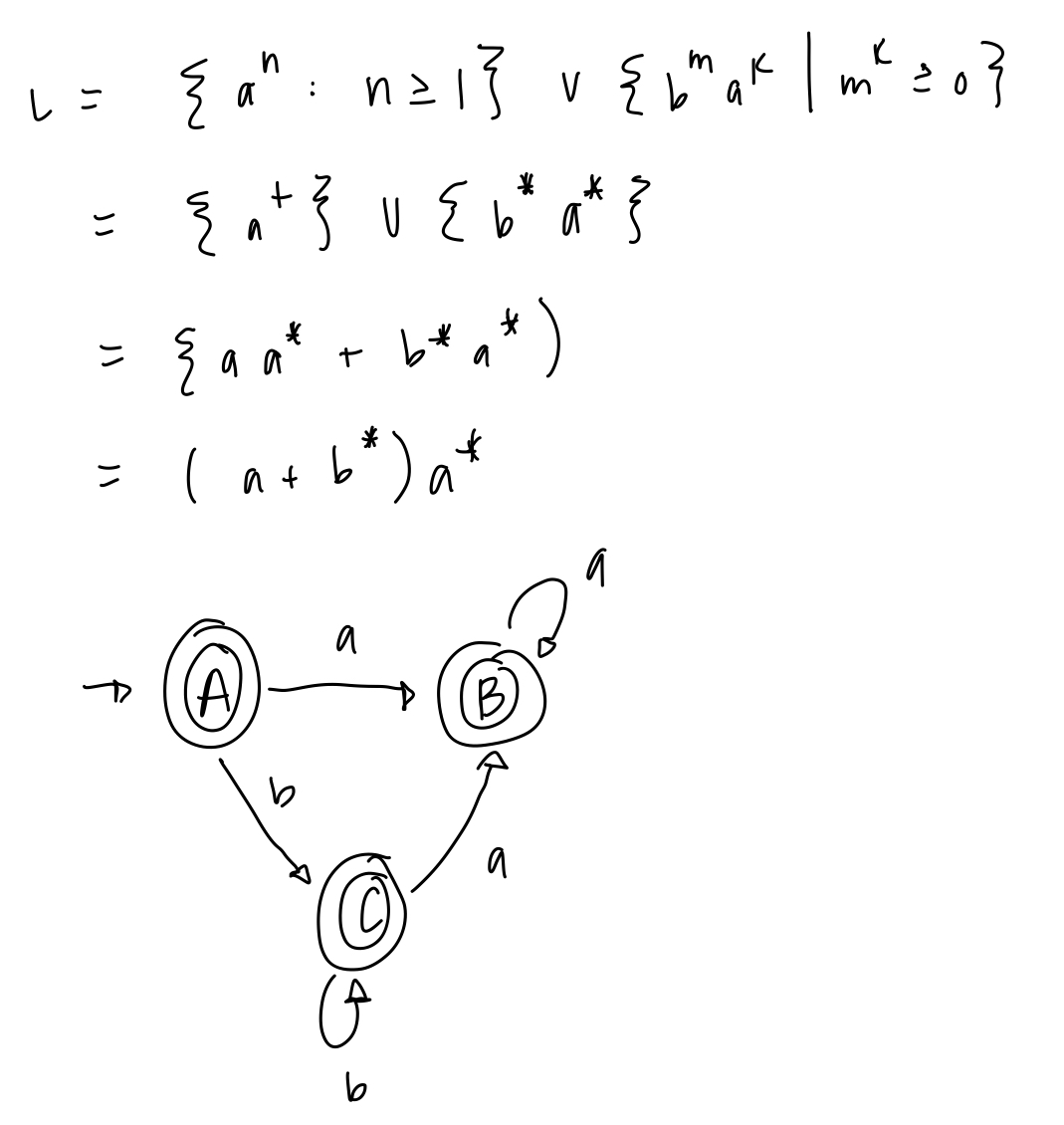


* See the attached solution



Q3. [8/15] For the language L = {*an* | *n* ³ 1 } È {*bmak* | *m* ³ 0, *k* ³ 0}

1. [8/8] Construct an NFA with three states that accepts L.



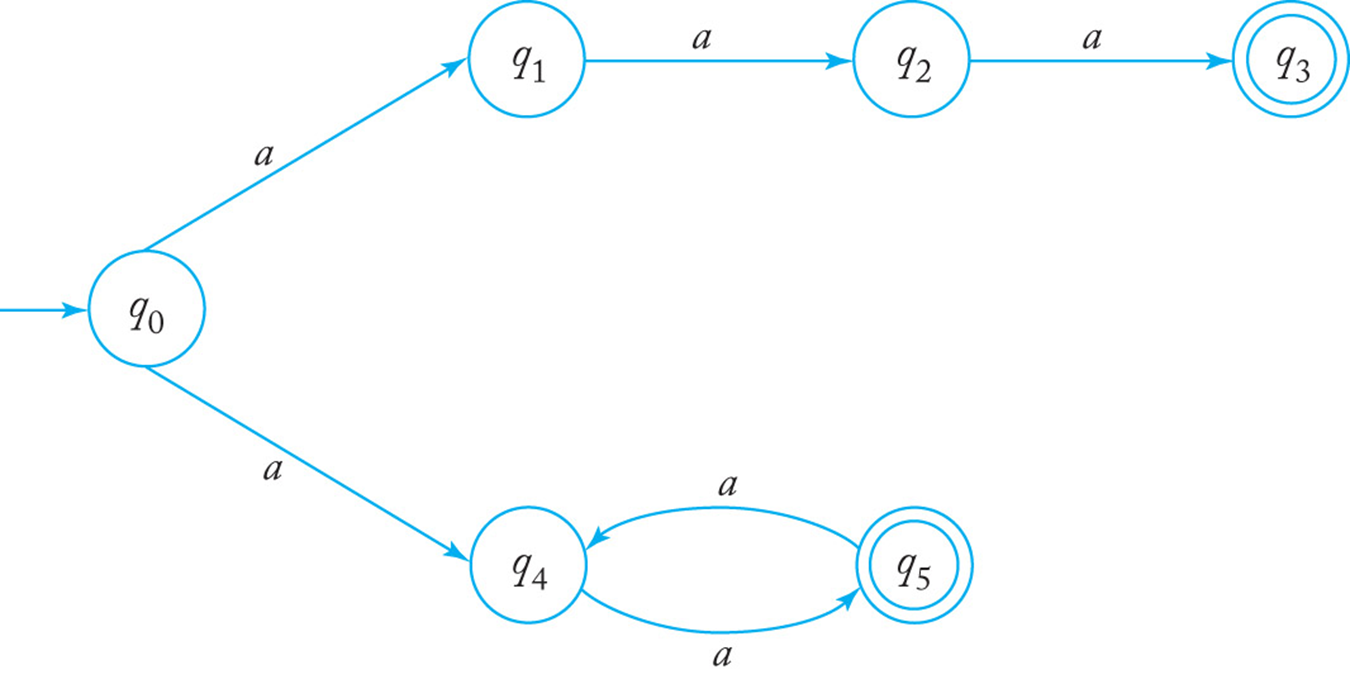
1. [0/7] Can you construct an NFA with the fewer states that accepts L? If so, construct it; otherwise, justify why your NFA in 1) is the minimal NFA.
   1. No, it cannot be constructed in 2 states.

Since L = {*an* | *n* ≥ 1 } ∪ {*bmak* | *m* ≥ 0, *k* ≥ 0} = {*bmak* | *m* ≥ 0, *k* ≥ 0}, an NFA with two states is possible.

A picture containing drawing

Description automatically generated

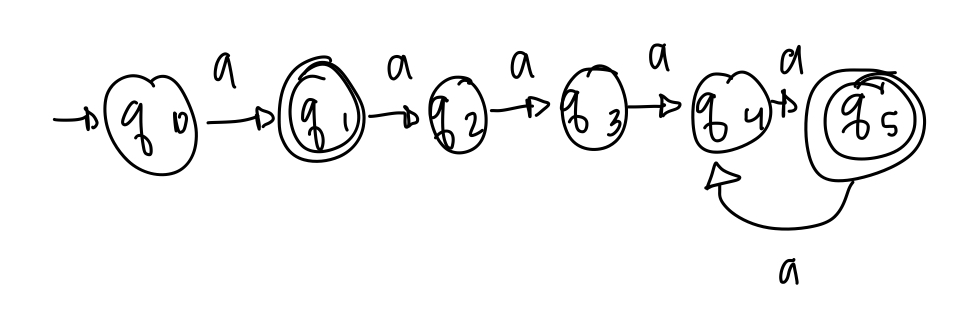
Q4. [17/20] For a given NFA in the figure,



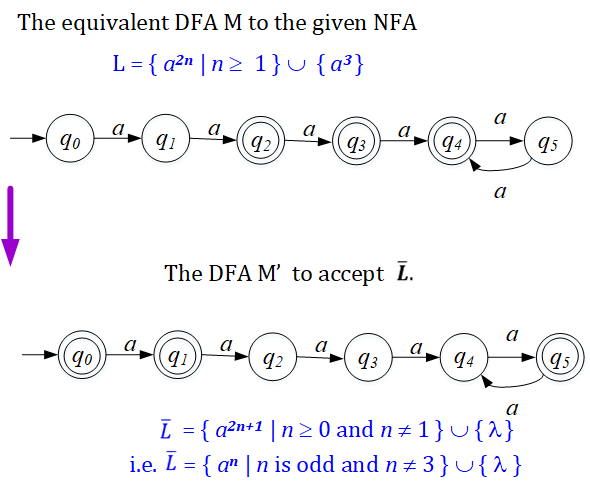
1. [8/10] Give a language *L* that is accepted by the NFA. Describe L in the proper mathematical format, not in the verbal English description. E.g.) L = { *a****n***| *n* ³ 0, *n* ¹ 3 }
   1. L = an | n = 3 OR n ≥ 2k, k ≥ 1

L = { *a****2n***| *n* ≥ 1 } ∪ { *a****3***} i.e. L = { *a****n***| *n* is even or *n* = 3 }

1. [9/10] Find a *DFA* that accepts the ***complement*** of the language defined by the NFA, i.e. .
   1. L = {an | a ≠ 3 AND n ≥ 2k +1, k ≥ 0

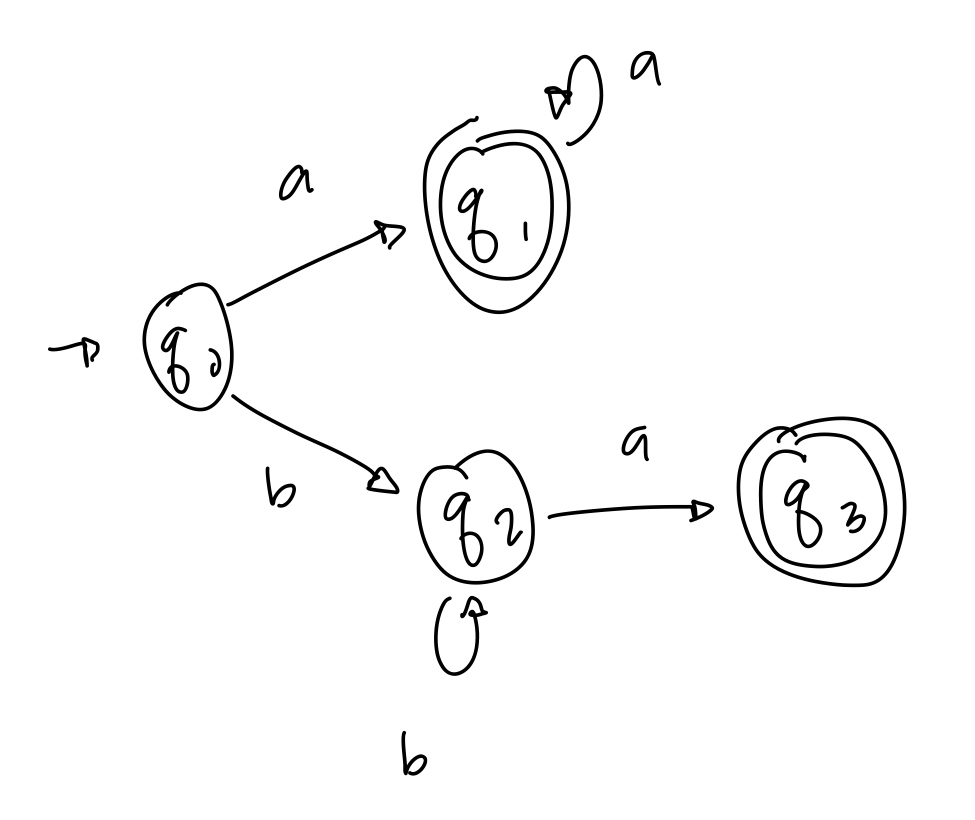


* q0 is also a final state.
* See the attached solution.



Q5. [9/10] Construct an NFA with the ***minimum*** number of states that accepts

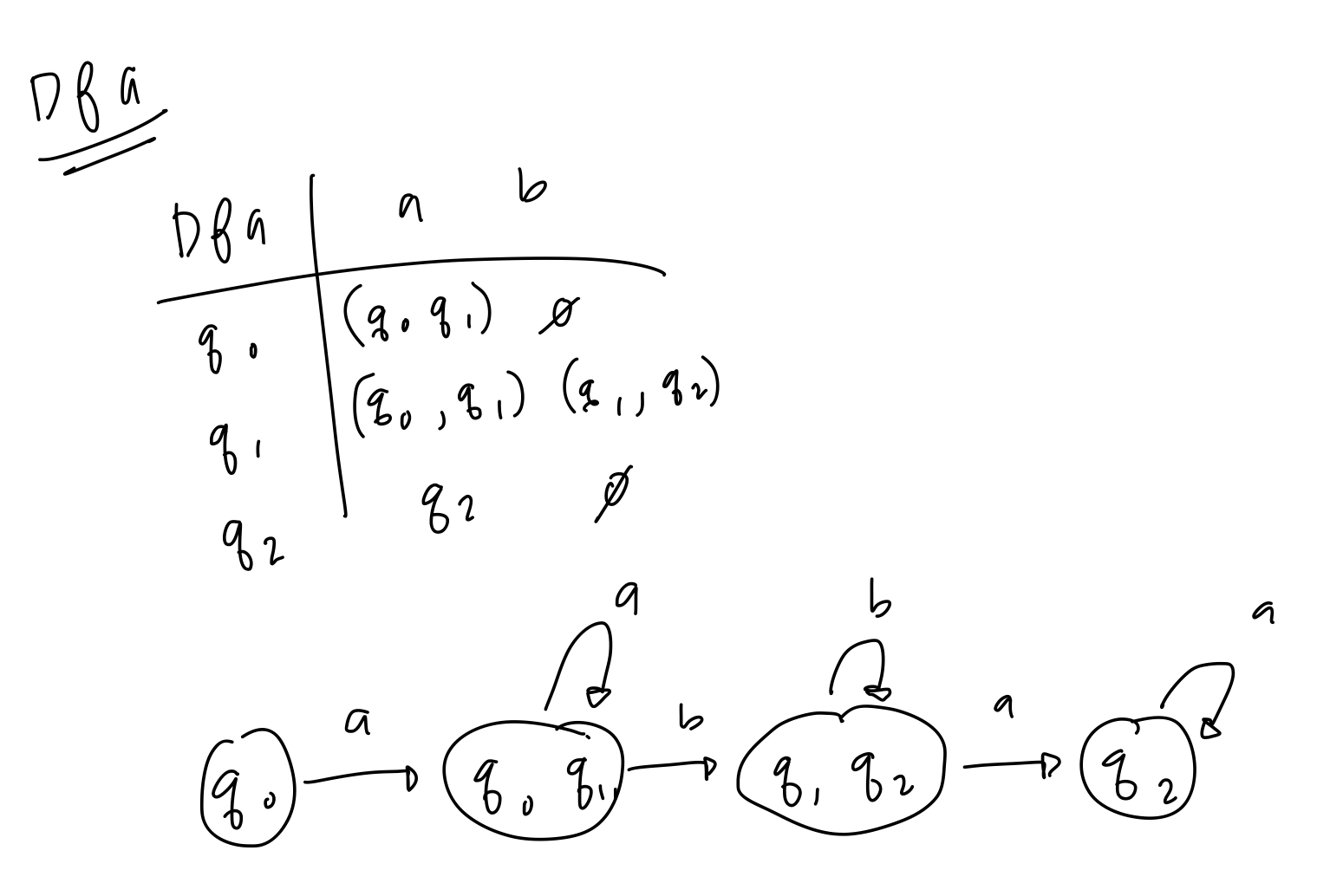
*L* = { *an* | *n* ³ 0 } È { *bna* | *n* ³ 1 }.



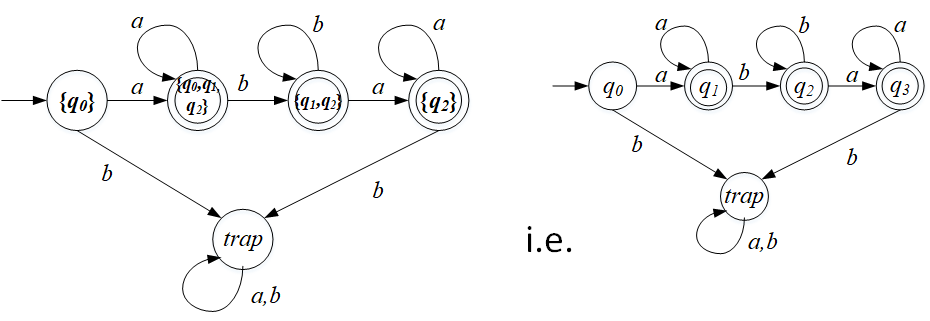
* q0 is also a final state.

Q6. [6/10] Convert the NFA defined by the transitions below with the initial state *q0* and the final state *q2* into an *equivalent DFA*. Draw the transition graph of the DFA.

d(*q0, a*) = { *q0, q1* }, d(*q1, b*) = { *q1, q2* }, d(*q2, a*) = { *q2* }, d(*q1,* l) = { *q1, q2* }.

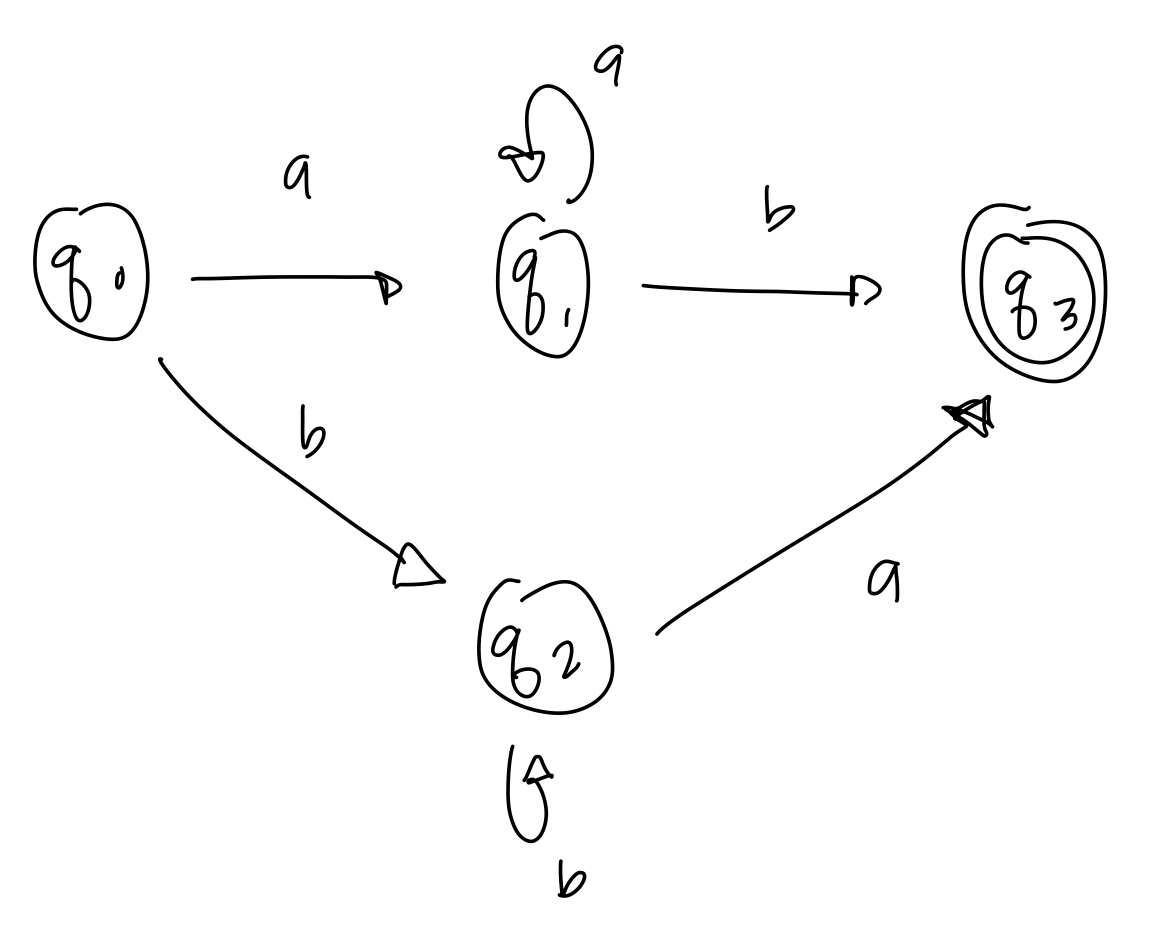


* See the attached solution



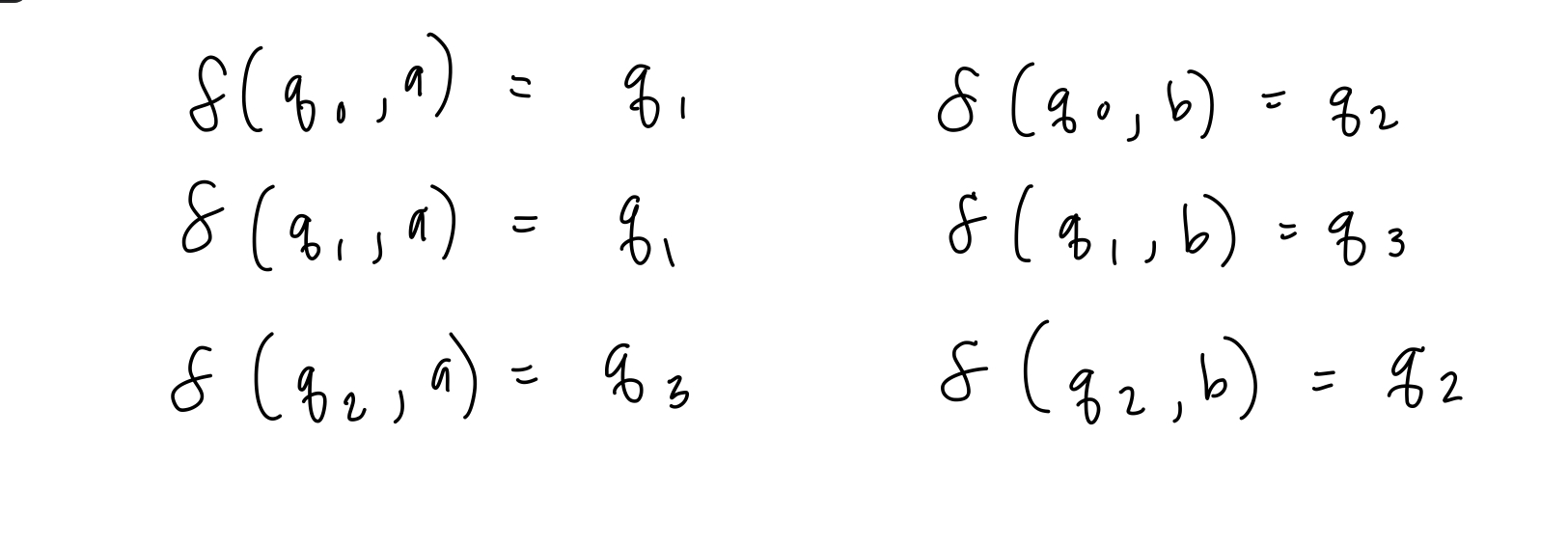
Q7. [20] For a given language, L = { *anb* | *n* ³ 1 } È { *bna* | *n* ³ 1},

1. [8/10] Construct a *minimal DFA* with the minimum number of states that accepts L.



* Since it is a DFA, q3 needs a transition to a trap state q4 with the transition of a, b. Similarly, q4 needs a transition to itself with a, b.

1. [8/10] Prove that your DFA in 1) is minimal. Hint: Check if any pair of the states are indistinguishable to be merged in the same class so that the number of states are minimized



* More explanation is expected

We claim it’s a minimal DFA.

Since q3 ∈ F and q4 ∉ F, q3 and q4 are distinguishable.

Since δ(q4, *a*) = q4 ∉ F and δ(q2, *a*) = q3 ∈ F, q2 and q4 are distinguishable.

Similarly, δ(q0, *b*) = q2 ∉ F and δ(q1, *b*) = q3 ∈ F, q0 and q1 are distinguishable.

Similarly, show that all the five states are mutually distinguishable.

Thus, the DFA is minimal.

Q8. [0/10, optional] Prove or disprove the following conjecture: If L is regular, so is LR.

If it is true, construct a NFA MR s.t. L(M’) = LR , from a NFA M that accepts L, i.e. L(M) = L. Then, show that L(M’ ) = LR .

Otherwise, give a counter example.

1. Since L is regular, there exists an NFA M that accepts L s.t. L = L(M) where M = (*Q*, Σ, δ, *q0*, *F* ),

To show LR is regular, let’s construct M’ that accepts LR as follows.

* + The start state *q0*, in *M* becomes the final state in *M’*.
  + Since there may be multiple final states in M, i.e. |F| ≥ 1, create a new start state p0  in M’ . Then, add a transition with λ from p0 to each of *qf* ∈ F.
  + The direction of all transition edges in *M* is reversed.
  + Thus, *M’* = (*Q*, Σ, δR, *p0’*, *q0* )

where ∃ (*qj, a*) = *qi* ∈ δR , ∀(*qi, a*) = *qj*∈δ

and (*p0,* λ) = *qf* for each *qf* ∈ F .

1. Then, show that L(M’) = LR .

→) Claim: For any *w∈ L(M’), w* ∈ *LR .*

Since *w∈ L(M’), w* is accepted by *M’,*

i.e. there is an transition from *p0* leading to the final state *q0* with *w* in M’ :

*δ R\* (p0, w) = δ R\* (p0, λw) = δ R\* (δ R (p0, λ), w) = δ R\* (qf* .*, w) = q0* for any *qf* ∈ F .

Since every transition in *M’* is the reverse of the transition in *M,*

for any *δ R\* (p0, w) = δ R\* (qf* .*, w) = q0 in M’,* there exists  *δ\* (q0, wR) = qf*  in *M.*

Thus, *wR ∈ L, i.e. w ∈ LR .*

←) Claim: For any *w* ∈ *LR , w∈ L(M’)*

For any *w* ∈ *LR , wR∈ L.*

Since *L* is a regular language accepted by *M, wR∈ L = L(M).*

So, there exists an extended transition *δ\* (q0, wR) = qf*  in *M.*

Since *M’* was defined with the reverse transitions of *M,*

*δ\* (q0, wRλ) =δ (δ\* (q0, wR ), λ)=δ (qf , λ)= δ\* (δ R (p0, λ), wR) = δ R\* (δ R (p0, λ), w)*

*=* *δ R\* (p0, λw) =* *δ R\* (p0, w) = q0 .* So, *w* ∈ *L(M’).*

Thus, L(M’) = LR .

Therefore, LR  is regular. Q.E.D.