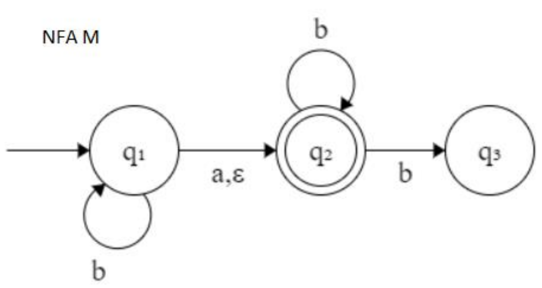
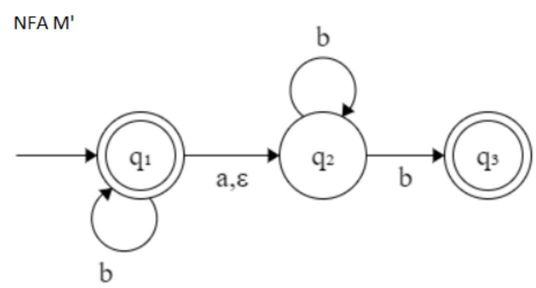
Problem 1: NFA Properties (20)

Show by giving an example that if *M* is an NFA that recognizes language *C*, swapping the accept and non-accept states in *M* **doesn't** necessarily yield a new NFA that recognizes the complement of *C*. Is the class of language recognized by NFAs closed by complement? Explain your answer. *Hint: Remember equivalence and its power, and implication and its limitations.*





If we assume NFA M with the $\Sigma = \{a, b\}$ and $Q = q_1, q_2, q_3$, and we create M' by swapping the accept states we can see that both M and M' accept the input "b", therefore swapping the accept and non-accept states in M doesn't necessarily yield a new NFA that recognizes a complement of C.

The class of language recognized by NFAs are closed by complement since DFAs are closed by complement and DFAs and NFAs recognize the same class of languages.

Problem 2: Regular Expression (15)

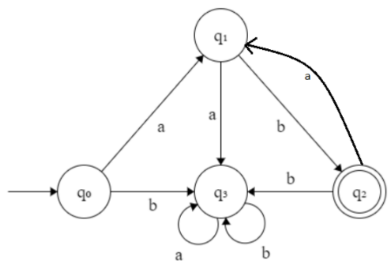
In certain programming languages, comments appear between delimiters such as **/#** and **#/**. Let *C* be the language of all valid delimited comment strings. A member of *C* must begin with **/#** and end with **#/** but have no intervening **#/**. For simplicity, assume the alphabet Σ for *C* is Σ = { **a**, **b**, **/**, **#** }. Give a regular expression that generates *C*.

$/\#(\#|((a|b)(a|b|/)*))*\#/$

Problem 3: The Regular Property (30)

With *A* and *B* as regular languages, show that the following languages **are** regular:

- *PSHUFFLE*(*A*, *B*) = $\{ s \mid s = a_1b_1a_2b_2 \ldots a_kb_k,\ a_1a_2 \ldots a_k \in A,\ b_1b_2 \ldots b_k \in B$, and each $a_i, b_i \in \Sigma \}$



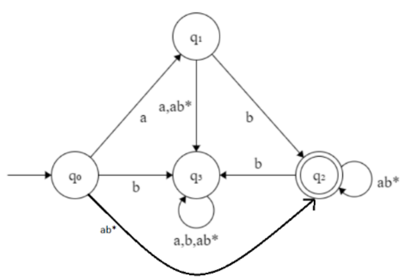
$Q = \{q_0, q_1, q_2, q_3\}$
$\Sigma = \{a, b\}$
$q_0 \in Q$
$F \subseteq Q \text{ where } F = \{q_2\}$
$\delta \text{ as shown by table}$

state	Input a	Input b
q_0	q_1	q_3
q_1	q_3	q_2
q_2	q_2	q_3
q_3	q_3	q_3

A DFA can be constructed for language PSHUFFLE(A,B), therefore it is a regular language. Also since A and B are regular and the Union of two languages are regular, then PSHUFFLE(A,B) is regular

- *SHUFFLE*(*A*, *B*) = { $s \mid s = a_1b_1a_2b_2 \ldots a_kb_k$, $a_1a_2\ldots a_k \in A$, $b_1b_2\ldots b_k \in B$, and each a_i, $b_i \in \Sigma^*$ }

A DFA can be constructed for language SHUFFLE(A,B), therefore it is a regular language. Also since A and B are regular and the union of two regular languages is still a regular and the concatenation of two regular languages is still regular then SHUFFLE(A,B) is regular since it's just the union of a and b and the concatenation of ab onto itself repeatedly

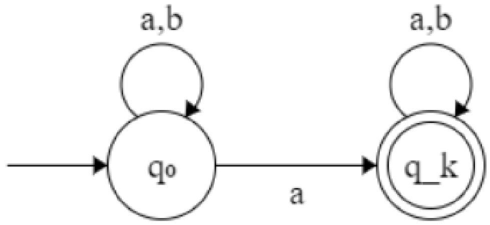


$Q = \{q_0, q_1, q_2, q_3\}$
$\Sigma = \{a, b, ab^*\}$
$q_0 \in Q$
$F \subseteq Q \text{ where } F = \{q_2\}$
$\delta \text{ as shown by table}$

state	Input a	Input b	Input ab^*
q_0	q_1	q_3	q_2
q_1	q_3	q_2	q_3
q_2	q_2	q_3	q_2
q_3	q_3	q_3	q_3

Problem 4: Parameterized Regularity (20)

Let Σ = { **a**, **b** }. For each positive integer *k*, let C_k be the language consisting of all strings that contain an **a** exactly *k* places from the right-hand end; in other words, C_k is all strings $\Sigma^*\mathbf{a}\Sigma^{k-1}$. Describe an NFA with *k* + 1 states that recognizes C_k in terms of **both** a state diagram **and** a formal description.



$Q = \{\ldots q_k\}$
$\Sigma = \{a, b\}$
$q_0 \in Q$
$F \subseteq Q$ *where* $F = \{q_k\}$
δ *as shown by table*

state	*Input a*	*Input b*
q_0	q_0, q_1	q_0
q_k	q_k	q_k

Problem 5: Irregularity (40)

Show that the following languages are not regular:

A. { *sss* | *s* is any string over Σ } with Σ = { **a**, **b** }

Assume the language is regular, with $pumping\ length\ = p$. Consider a pumping string $s = (ababab)^p$ So, we have $s = xyz$ with x the prefix, y the cycle string and z the suffix, so that:

$PL1: xy^iz \in L\ for\ all\ i \in N\ or\ xy^*z \in L$

$PL2: |y| > 0$ $PL3: |xy| \leq p$

By definition of our language, all xyz divisions of s must be divisible by some substring s 3 times since our pumping string is some substring s concatenated by itself twice

- $by\ PL1$, set $i = 2$ and consider $xy^iz = xyyz$
  - But $xyyz$ cannot be divided by some substring s 3 times since $xyyz$ has more letters than xyz
  - Then by definition of $L, xyyz \notin L, \rightarrow\leftarrow PL1$

So, for every possible construction of $\boldsymbol{y}$, $\boldsymbol{xyyz \notin L}$, →← PL1. By contradiction, the language must be irregular.

B. { *x* = *y* + *z* | *x*, *y* and *z* are binary integers, and *x* is in fact the sum of *y* and *z* } with Σ = { **0**, **1**, **+**, **=** }

Assume the language is regular, with $pumping\ length\ = p$. Consider a pumping string $s\ = (1^p = 0 + 1^p)$. So, we have $s = xyz$ with x the prefix, y the cycle string and z the suffix, so that:

$PL1: xy^iz \in L\ for\ all\ i \in N\ or\ xy^*z \in L$

$PL2: |y| > 0$ $PL3: |xy| \leq p$

By definition of $PL1$, $PL2$ and our pumping string s, all xyz divisions of s must be:

- $x = 1^a$
- $y = 1^b$, where b must be greater than 0
- $z = 1^{p-(a+b)} = 0 + 1^p$
  - Since $|xy| \leq p$, the first 1^p must be made of x and y with at least the last 1 being a part of z.
  - If the first 1^p were made of only x and y we wouldn't be able to have $xyyz$ since that would mean $|xy|$ would exceed length p
- So, the resulting string xyz should be equal to $1^a 1^b 1^{p-(a+b)} = 0{+}1^p = 1^p = 0 + 1^p$
- by $PL1$, set $i = 2$ and consider $xy^i z = xyyz$
  - Since $xyyz = (1^a 1^b 1^b 1^{p-(a+b)} = 0{+}1^p)$ which simplified is $(1^{b+p} = 0 + 1^p)$
  - And $1^{b+p} \neq 1^p$ which violates our language
  - Then by definition of $L, xyyz \notin L, \rightarrow\leftarrow PL1$

So, for every possible construction of $\boldsymbol{y}$, $\boldsymbol{xyyz \notin L}$, →← PL1. By contradiction, the language must be irregular.

C. { $\mathbf{a}^n\mathbf{b}^m\mathbf{a}^n$ | *m* and *n* are non-negative integers } with Σ = { **a**, **b** }

Assume the language is regular, with $pumping\ length\ = \ p$. Consider a pumping string $s = a^p b^m a^p$. So, we have s = xyz with x the prefix, y the cycle string and z the suffix, so that:

$PL1: xy^i z \in L\ for\ all\ i \in N\ or\ xy^* z \in L$

$PL2: |y| > 0$ $PL3: |xy| \leq p$

By definition of $PL1$, $PL2$ and our pumping string s, all xyz divisions of s must be:

- $x = a^a$
- $y = a^b$, where b must be greater than 0
- $z = a^{p-(a+b)} b^m a^p$, z must have the remaining a's in the first a^p followed by a b and then a^p
  - Since $|xy| \leq p$, the first a^p must be made of x and y with at least the last a being a part of z.
  - If the first a^p were made of only x and y we wouldn't be able to have $xyyz$ since that would mean $|xy|$ would exceed length p

- So, the resulting string xyz should be equal to $a^a a^b a^{p-(a+b)} b^m a^p = a^p b^m\ a^p$
- by $PL1$, set $i = 2$ and consider $xy^i z = xyyz$
  - Since $xyyz = a^a a^b a^b a^{p-(a+b)} b^m a^p = a^{b+p} b^m a^p$
  - And $a^{b+p} \neq a^p$ which violates our language
  - Then by definition of $L, xyyz \notin L, \rightarrow\leftarrow PL1$

So, for every possible construction of $\boldsymbol{y}$, $\boldsymbol{xyyz \notin L}$, →← PL1. By contradiction, the language must be irregular.

D. { s | s is a string over Σ that is not a palindrome } with Σ = { **a**, **b** }

Assume the complement language is regular, with $pumping\ length\ = p$. Consider a pumping string $s = a^p b a^p$. So, we have s = xyz with x the prefix, y the cycle string and z the suffix, so that:

$PL1{:}\, xy^i z \in L\ for\ all\ i \in N\ or\ xy^* z \in L$

$PL2{:}\, |y| > 0\ PL3{:}\, |xy| \leq p$

By definition of $PL1$, $PL2$ and our pumping string s, all xyz divisions of s must be:

- $x = a^a$
- $y = a^b$, where b must be greater than 0
- $z = a^{p-(a+b)} b a^p$, z must have the remaining a's in the first a^p followed by a b and then a^p
  - Since $|xy| \leq p$, the first a^p must be made of x and y with at least the last a being a part of z.
  - If the first a^p were made of only x and y we wouldn't be able to have $xyyz$ since that would mean $|xy|$ would exceed length p
- So, the resulting string xyz should be equal to $a^a a^b a^{p-(a+b)} b a^p = a^p b a^p$
- by $PL1$, set $i = 2$ and consider $xy^i z = xyyz$
  - Since $xyyz = a^a a^b a^b a^{p-(a+b)} b a^p = a^{b+p} b a^p$
  - And the reverse, or $(xyyz)^R = a^p b a^a a^b a^b a^{p-(a+b)} = a^p b a^{b+p}$
  - Since $a^{b+p} b a^p \neq a^p b a^{b+p}$ violates our language
  - Then by definition of $L, xyyz \notin L, \rightarrow\leftarrow PL1$

Therefore, for every possible construction of $\boldsymbol{y}$, $\boldsymbol{xyyz \notin L}$, →← PL1 and by contradiction, the complement language must be irregular. And if the complement language is irregular then our original language is irregular because, by definition, regular languages are closed under complement.