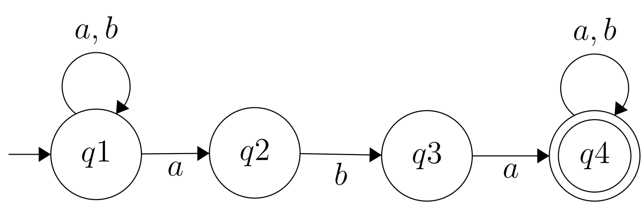
On my honor, I have not given, nor received, nor witnessed any unauthorized assistance on this work.

Print name and sign: ______________________________

Question:	1	2	3	Total
Points:	15	7	8	30
Score:				

1. Consider the following NFA defined with $\Sigma = \{a, b\}$:



(a) State whether or not each string is accepted by the NFA:

i. (1 point) `ab` ____no____

ii. (1 point) `aaabaaa` ____yes____

iii. (1 point) `abbb` ____no____

iv. (1 point) `abbba` ____no____

(b) (1 point) Give an informal description of the strings accepted by this NFA.

Solution: Set of all strings containing the “aba” substring.

(c) (3 points) Give the formal 5-tuple definition for the NFA.

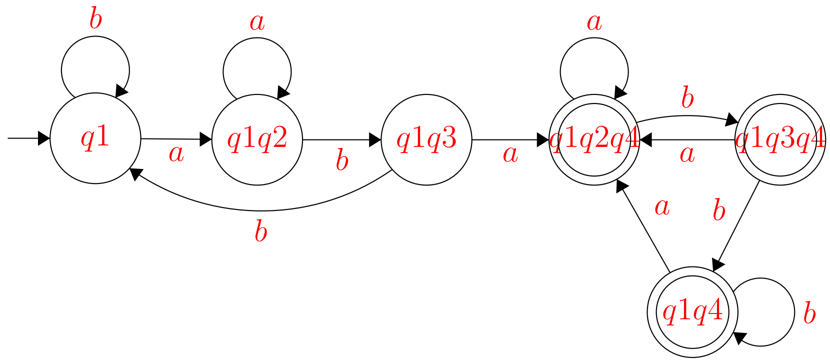
Solution: $M = (\{q1, q2, q3, q4\}, \{a, b\}, \delta, q1, \{q4\})$
where $\delta =$

state	a	b
q1	{q1,q2}	{q1}
q2	$\emptyset$	{q3}
q3	{q4}	$\emptyset$
q4	{q4}	{q4}

(d) (7 points) Draw an equivalent DFA. Be sure to show your transition table as well.

Solution:

state	a	b
q1	q1q2	q1
q1q2	q1q2	q1q3
q1q3	q1q2q4	q1
q1q2q4	q1q2q4	q1q3q4
q1q3q4	q1q2q4	q1q4
q1q4	q1q2q4	q1q4



2. (7 points) Consider the language of *palindromes* (that is, strings which are the same whether read forwards or backwards) over the alphabet $\Sigma = \{0, 1\}$.[1] We want to use the pumping lemma to show that this language is not regular. The proof is begun for you below. Finish it (and remember not to make any leaps of logic or faith).

Suppose that the set of palindromes were regular. Let p be the pumping length. Consider the string $s = 0^p110^p$. s is clearly a palindrome and $|s| >= p$ so...

Solution: ...by the pumping lemma, there must exist strings x, y, and z satisfying the four constraints of the pumping lemma.

So, pick any x, y, and z such that $s = xyz, |xy| <= p$, and $|y| > 0$. Because $|xy| <= n$, xy is entirely contained in the 0^p at the start of s. That leaves us with 2 cases:

Case 1: x and y contain some (but not all) of the initial 0's.

So x and y consist entirely of zeros, i.e. x = 0^m and y = 0^n where $m + n < p$. Then z must look like $0^{p-m-n}110^p$. Then, for i = 1, everything is fine and the resulting string: $0^m0^n0p - m - n110^p$ is a palindrome.

However, if i = 2, then our string becomes $0^m0^{2n}0^{p-m-n}110^p$ which is not a palindrome as there are too many 0's at the beginning of the string ($m + 2n + p - m - n == p + n$) and only p 0's at the end of the string. Thus, this pumped string is not in L.

Case 2: x and y together contain all of the initial 0s.
So x and y consist entirely of zeros, i.e. x = 0^m and y = 0^n where $m + n = p$. Then z must look like 110^p. Then, for i = 1, everything is fine and the resulting string: $0^m0^n110^p$ is a palindrome as $m + n = p$.

However, if i = 2, then our string becomes $0^m0^{2n}110^p$ which is not a palindrome as there are too many 0's at the beginning of the string ($p + n$) and only p 0's at the end of the string. Thus, this pumped string is not in L.

This means that the set of palindromes doesn't satisfy the pumping lemma and, thus, the set of palindromes cannot be regular.

Common mistakes/problems:

- Picking a specific value of p (like 2) and then using a specific example of a string like 001100 and showing there was a way to pump this to make it not a palindrome. This is a good way to intuitively understand how the pumping lemma could be used, but for a proof you must generalize your concepts to work for any value of p.

[1] Concrete examples of palindromes would be 00011000 or 010010 while 1000 or 010101 would not be palindromes.

- Proving only 1 of the two cases above. Both are needed.
- Several students presented cases where x or y contained some of the 1's. This violates the pumping lemma because $|xy > p$ so it's not a valid dividing of the string.
- Several students stated a sequence of mathematical steps without "gluing" them together with logic. Make sure one step follows logically from another clearly and methodically.

3. (8 points) Draw an NFA for the regular expression $(a^*|(ab)^*)b^*$. If you use any abstractions (eg. black box parts of your NFA), make sure the states and transitions can be clearly understood.

Solution: This problem is pretty straightforward because the Kleene star operator is on each piece. The only real mistake I saw was not concatenating b^* but treating it as if it was $a^*|(ab)^*|b^*$.