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

Name & Signature: ____________________

Name & Signature: ____________________

Name & Signature: ____________________

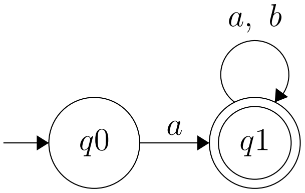
Name & Signature: ____________________

Name & Signature: ____________________

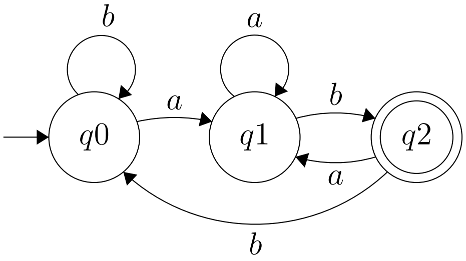
Question:	1	2	3	4	5	6	Total
Points:	12	33	20	10	10	15	100
Score:							

1. Topic: Recognizing and identifying NFAs
   For each of the following machines, state whether it is a DFA or NFA, and give a reason for your classification. Then informally describe the language accepted by the DFA/NFA.

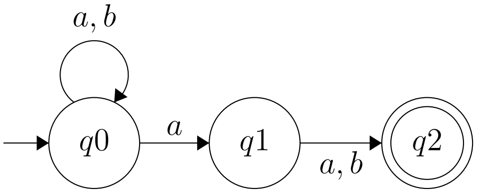
   (a) (3 points) $\Sigma = \{a, b\}$



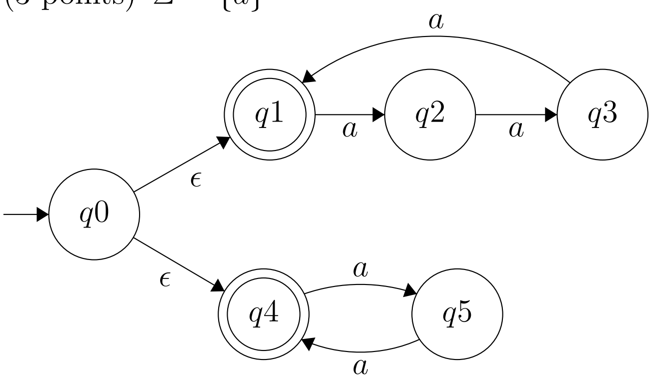
   (b) (3 points) $\Sigma = \{a, b\}$



   (c) (3 points) $\Sigma = \{a, b\}$



(d) (3 points) $\Sigma = \{a\}$



2. Topic: Constructing NFAs using DFAs
   Given $\Sigma = \{0, 1\}$, let L_1 be the set of all strings that start with a 0. Let L_2 be the set of all strings that end with a 0.

   (a) (10 points) Draw an DFA for both L_1 and L_2

(b) (4 points) Give a regular expression for both L_1 and L_2

(c) Using your DFAs from part a, construct an NFA for L_3 and give a regular expression for it. You do not necessarily need to redraw your DFAs in their entirety. You can draw boxes labeled L_1 and L_2 if that is appropriate in your answer.

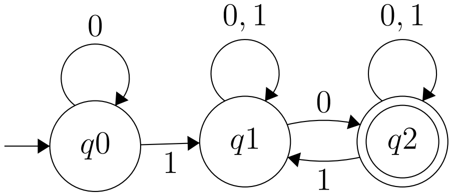
i. (6 points) $L_3 = L_1 \bullet L_2$ (concatenation)

ii. (6 points) $L_3 = L_1 \cup L_2$

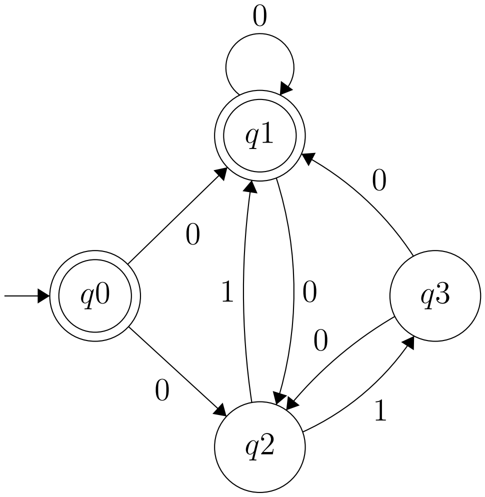
iii. (7 points) $L_3 = (L_1 \cup L_2)^*$

3. Topic: Equivalence of NFAs to DFAs.
   Convert the following NFAs to DFAs. Show the transition table you build during the conversion. You do not need to minimize the resulting DFA.

   (a) (10 points) $\Sigma = \{0, 1\}$



   (b) (10 points) $\Sigma = \{0, 1\}$


0
q1
0
0
q0
1
0
q3
0
0
1
q2

4. (10 points) Topic: Regular Expressions and NFAs.
Construct an NFA which accepts $L((ab|a)^*(bb))$. You do not need to minimize or reduce states. *Hint: build NFAs for small pieces of the regular expression and then combine those small pieces to build a bigger NFA.*

5. (10 points) Use the pumping lemma to show that the following language is not regular:

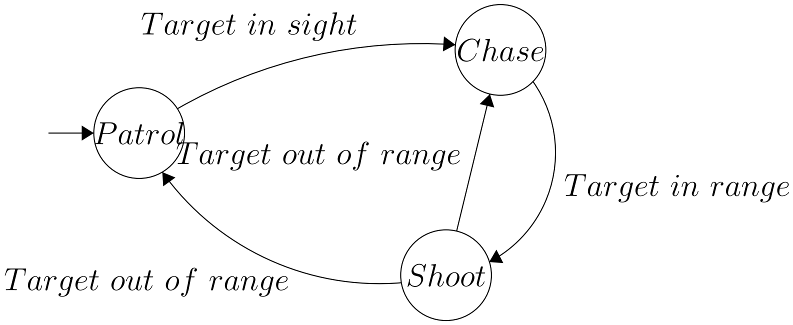
$$L = \{0^a 1^b 0^{a+b} : a, b \geq 0\}$$

Be very clear and concise in your answer/proof. For full credit, I must be able to follow your argument without leaps of logic (or faith).

6. (15 points) Application: Video games
A computer controlled character can be modeled by a finite state machine. If we used a DFA to model behavior, the same inputs would lead to the same outcome each time. Booooring. But by introducing non-determinism to the game, we can make things interesting. While not the most sophisticated way of generating behaviors in a game, many early games (think Pac-Man, etc) were modeled by finite state machines.

Design an NFA for a computer controlled basketball, football, soccer, or other team-sport player in a competitive game. Pick a sport that most people on your team feel comfortable discussing, but explaining the game to someone who is unfamiliar with it can actually be a great way to begin identifying states and transitions. Your machine should have at least 5 states and be non-deterministic. Provide a brief narrative of how your machine works.

Below is an example NFA for a generic ”soldier” non-player character in a game and a brief narrative clarifying the machine.



Patrol state: solider moves back and forth randomly until it encounters a target. It then transitions to the Chase state until it moves close to the target. At that point, we transition to the Shoot state and attempts to shoot the target. If the target moves out of shooting range, the solider will either return to patrolling or give chase to the target.