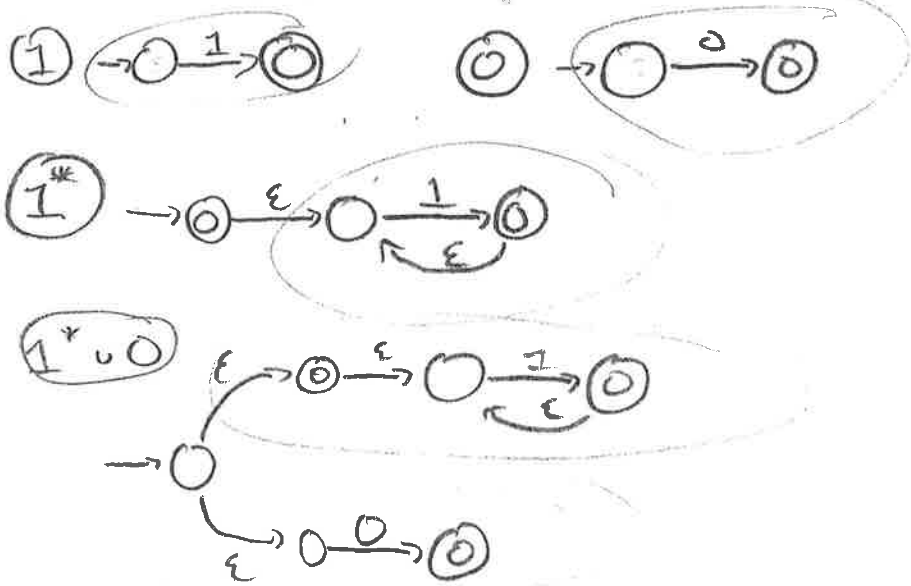
$\frac{70}{60} = \frac{117}{100}$ 20/20

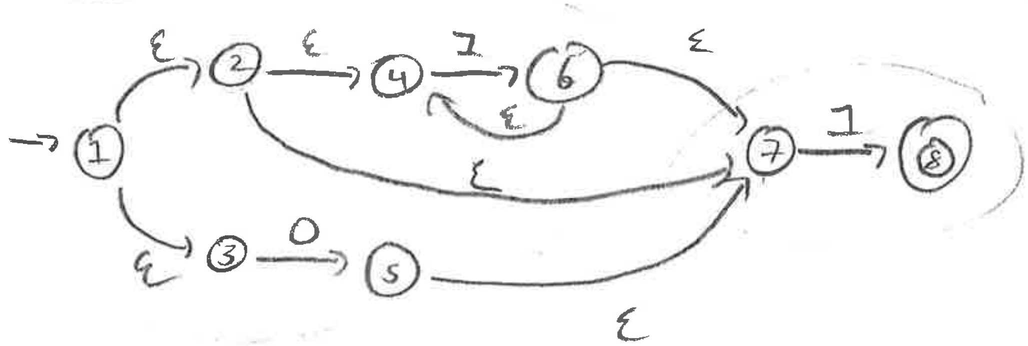
CS 3350 Automata, Computability, and Formal Languages
Spring 2016, Test 1

Name: [redacted]

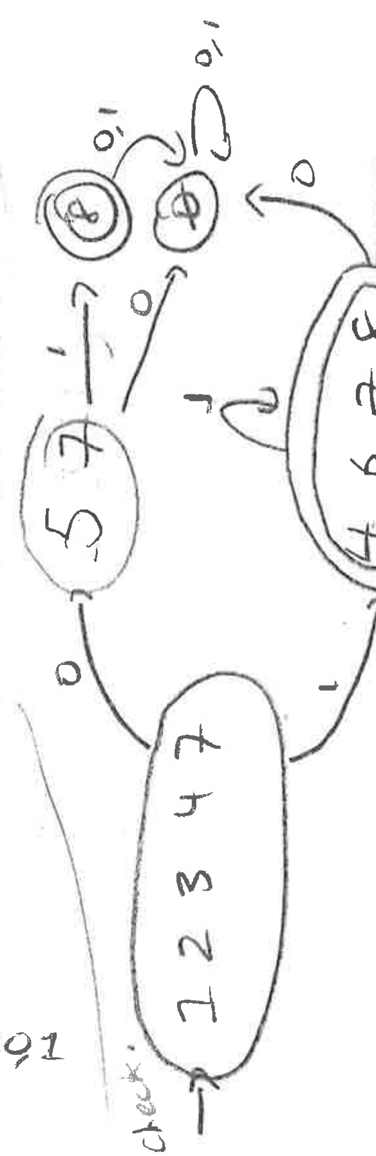
1-2. Use a general algorithm to design a non-deterministic finite automaton recognizing the language $(1^* \cup 0)1$. After that, use the general algorithm to design a deterministic finite automaton recognizing this same language.



$(1^* \cup 0)1$ NDFA

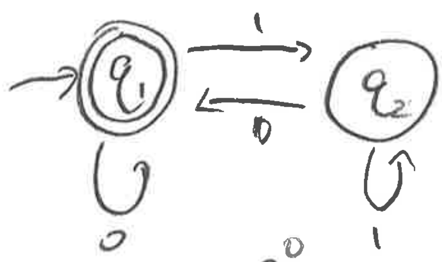


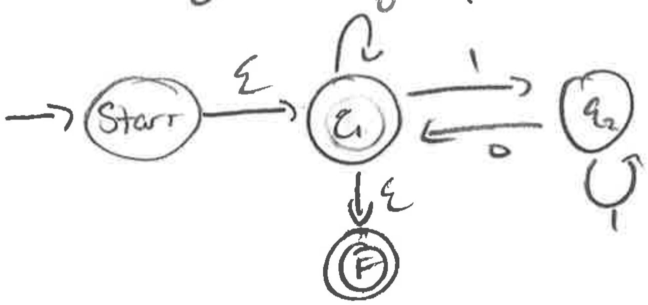
DFA $(1^* \cup 0)1$

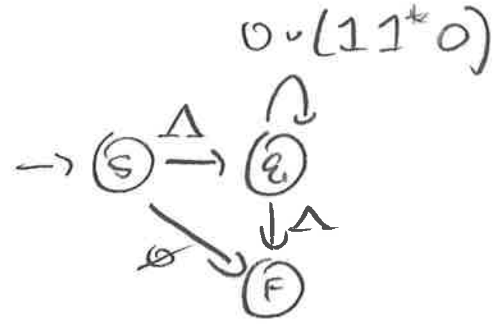


10/10

3. Use a general algorithm to transform the following finite automaton into the corresponding regular expression. This automaton has two states: a state q1 which is both a start state and a final state, and a state q2. In the state q1, 1 leads to q2, and 0 leads to q1. In the state q2, 1 leads to q2, and 0 leads to q1.







eliminate q_2:

$$D'_{S1} = \overset{\Lambda}{D_{S1}} \cup (\overset{\emptyset}{D_{S2} D^*_{22} D_{21}}) = \Lambda$$

$$D'_{SF} = \overset{\emptyset}{D_{SF}} \cup (\overset{\emptyset}{D_{S2} D^*_{22} D_{21}}) = \emptyset$$

$$D'_{11} = \overset{0}{D_{11}} \cup (\overset{1}{D_{12}} \overset{1^*}{D^*_{22}} \overset{0}{D_{21}}) = 0 \cup (11^*0)$$

$$D'_{1F} = \overset{\Lambda}{D_{1F}} \cup (\overline{D_{12} D_{22} \overset{\emptyset}{D_{2F}}}) = \Lambda$$

eliminate q_1:

$$D'_{SF} = \overset{\emptyset}{D_{SF}} \cup (\overset{\Lambda}{D_{S1}} \overset{(0\cup(11^*0))^*}{D^*_{11}} \overset{\Lambda}{D_{1F}}) = \boxed{(0 \cup (11^*0))^*}$$

10/10

4. On the example of the automaton from Problem 3, explain, in detail, how the sequence 010100 will be presented as xyz according to the pumping lemma. For this sequence, check -- by tracing step-by-step -- that the sequence xy^iz for $i = 2$ is indeed accepted by the automaton.

$P = 6$

$S = 010100$

$len(s) \geq p$ ✓

1 1 2 1 2 1 1 ✓

$\underbrace{\Lambda}_{x}|\underbrace{0}_{Y}|\underbrace{1|0|1|0|0|}_{z}$

$X = \Lambda$

$Y = 0$

$Z = 10100$

$len(y) > 0$ ✓

$len(xy) \leq P$ ✓

xy^2z: $\overset{x}{\Lambda}|\overset{y_1}{0}|\overset{y_2}{0}|\overbrace{1|0|1|0|0|}^{z}$

1 1 1 2 1 2 1 1 ✓

10/10

5. Use the Pumping Lemma to prove that the language L consisting of all the words of the type *waw* is not regular, where *a* is a letter and *w* can be any word. Here:

- if *w* is an empty string, we get the word *a*,
- if *w* is *ab*, we get *abaab*,

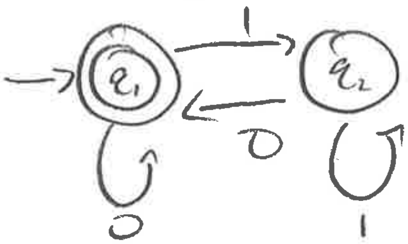
etc.

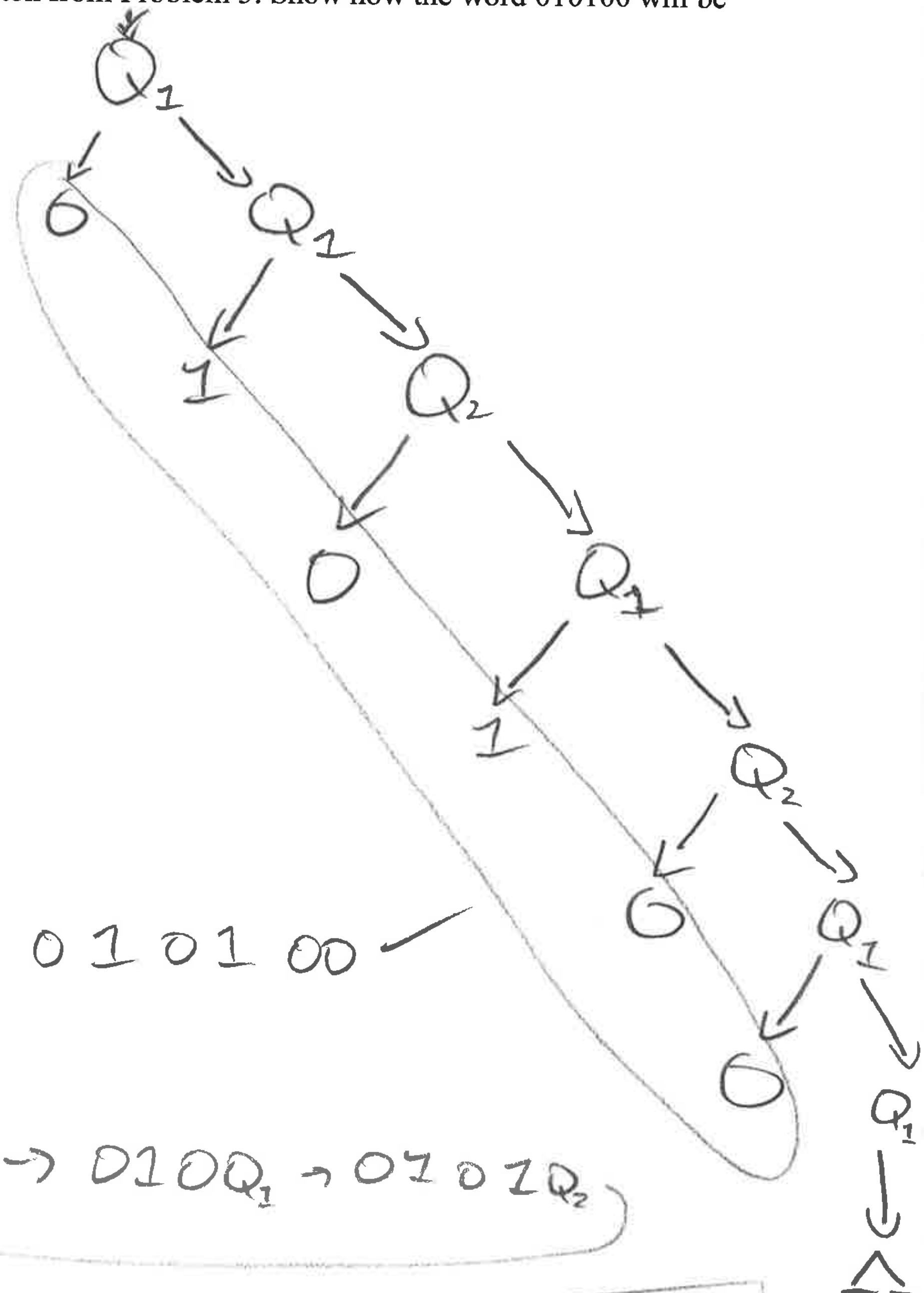
Proof by contradiction. Assume L is regular. If L is regular, there exists some number p for all words accepted by L such that the length of the word is ≥ p, and there exists some concatenation of xyz such that this concatenation represents the word, the length of xy does not exceed p, y is > 0, xy^iz is accepted by L for all i, and xy is found amongst the first p symbols.

Consider words of the form b^pab^p, a word that is accepted by L. The length of the word is 2p+1, which is ≥ p. However, since x & y must be found amongst the first p symbols, y must necessarily consist only of the letter b. Thus, xy^iz for i > 1 results in b^qab^p, where q is > p. This word is not accepted by L, which we assumed to be regular. However, by the Pumping Lemma, b^qab^p must be accepted by L. This is a contradiction, and thus, L is not regular. ∎

10/10

6. Use the general algorithm that we had in class to design a context-free grammar which generates exactly the words accepted by the automaton from Problem 3. Show how the word 010100 will be generated by this grammar.





RULES

$Q_1 \to 0Q_1$

$Q_1 \to 1Q_2$

$Q_2 \to 1Q_2$

$Q_2 \to 0Q_1$

$Q_1 \to \Lambda$

alt:

$Q_1 \to 0Q_1 \to 01Q_2 \to 010Q_1 \to 0101Q_2 \to 01010Q_1 \to 010100Q_1 \to \boxed{010100}\Lambda$

10/10

7. (For extra credit) Use the general algorithm to transform the grammar from Problem 6 into a pushdown automaton.

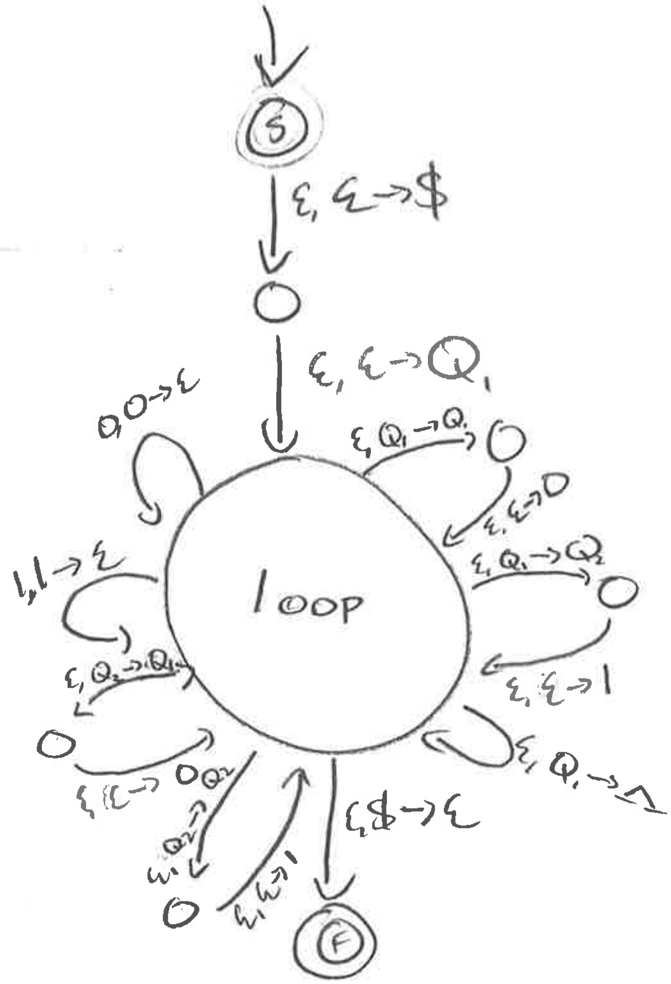
$Q_1 \to 0Q_1$

$Q_1 \to 1Q_2$

$Q_2 \to 0Q_1$

$Q_2 \to 1Q_2$

$Q_1 \to \Lambda$



Trace

0 1 0 1 0 0