Theory of Computation
Practice Midterm Solutions

Name: ________________________________

Directions:

Answer the questions as well as you can. Partial credit will be given, so show your work where appropriate. Try to be precise in your answers in order to maximize your points. Also make sure that your answers to pumping lemma questions are sufficiently clear so that I can tell that your reasoning is correct. Good luck.

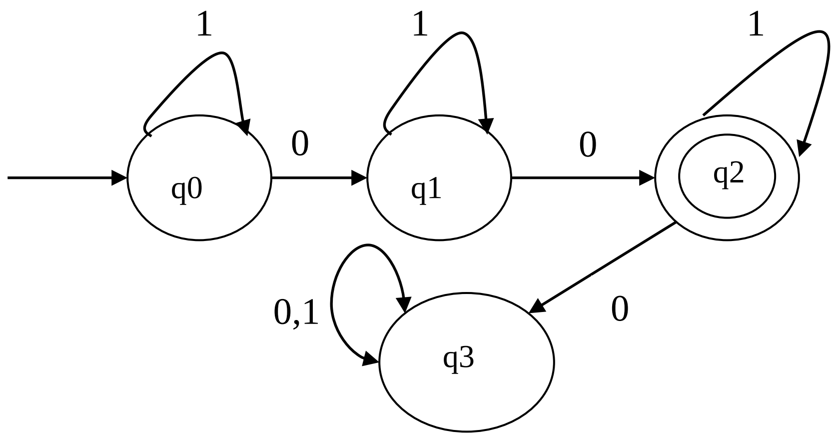
Note: DFA = Deterministic Finite Automata NFA = Nondeterministic Finite Automata
PDA = Push-Down Automata CFG=Context Free Grammar

Here are the pumping lemmas:

If A is a regular language, then there is a number p (the pumping length) where, if s is any string in A of length at least p, then s may be divided into 3 pieces, $s = xyz$, satisfying the following conditions:

1. For each $i \geq 0$, $xy^iz \in A$,
2. $|y| > 0$, and
3. $|xy| \leq p$

1. Let M be the Deterministic Finite Automata (DFA) shown below



a. Provide a <u>formal</u> description of M below (6 points)

M = ({q0, q1, q2, q3}, {0, 1}, δ, q0, {q2}) or alternatively

Q = {q0, q1, q2, q3}
Σ = {0, 1}
q0 = q0
F = {q2}

In either case, you need to specify δ as:

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

b. In plain English, describe the language described by this DFA. In order to get full credit, you need to provide reasonably succinct description that ignores irrelevant considerations. Fill in the blank below. (3 points)

The DFA shown above describes a language that <u>contains exactly 2 0's</u>.

c. Is the language described in part 1b a context-free language? Circle "Yes" or "No" and then briefly justify your answer. (3 points) $\boxed{\text{Yes}}$ No

Because the language is defined by a DFA M, it is a regular language. A regular language is always a context free language. That is because a CFL is defined by a PDA and a PDA is a NFA with a stack—so it includes all NFA's.

2. Actual Factuals. Answer the following True/False and short answer questions.

   a. A DFA is equivalent in expressive power to an NFA. $\boxed{\text{True}}$ False

   b. Deterministic and non-deterministic PDA's have equivalent expressive power. True $\boxed{\text{False}}$

   c. An NFA can recognize any language that a PDA can recognize True $\boxed{\text{False}}$

   d. You can build a PDA to recognize $\{0^{500}1^{40000} \cup 1^{1000}0^{200}\}$ $\boxed{\text{True}}$ False

   e. The language 0^n0^n is regular $\boxed{\text{True}}$ False

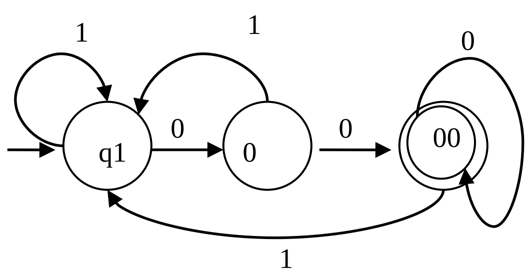
   f. You can convert an NFA with n states to a DFA with ____2^n_____ states.

   g. The minimum pumping length of the string 0*1* is: ______1_______

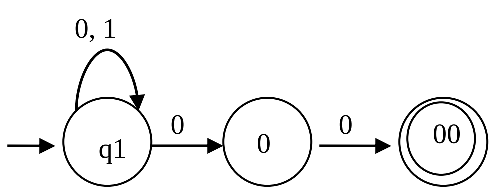
   h. The minimum pumping length of the string 00* is: ______0_______

3. Assume an alphabet Σ that is {0, 1}

   a. Draw the *simplest* possible DFA (in terms of number of states and arcs) that describes the language of all strings that end in "00". (7 points)



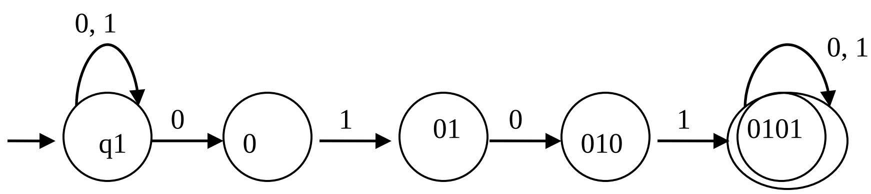
   b. Draw the *simplest* possible NFA (in terms of number of states and arcs) that describes the language of all strings that end in "00". (7 points)



   c. Provide the regular expression that describes the language in part a. (3 points)

   Σ^*00

4. Draw the NFA that recognizes the language where w contains the substring 0101. Do this using 5 states and assuming a binary alphabet.



5. Your friend Brian is trying to prove that the language ww^{-R}, the language of palindromes, is not regular. For pumping length *p* he chooses the string $S = 01^p1^p0$, which is a palindrome. Can he use the pumping lemma for regular languages to prove that this language is not regular?

   Circle one: Yes $\boxed{\text{No}}$

   Now explain why below:

   Although the language is not regular, you can find an assignment of xyz that can be pumped. Try the following:

   $x = 0, y = 1, z = 1^{p-1}1^p0$

   When we pump it, we get something of the form 01^x0, *where* $x \geq 2p$. *This is always a palindrome (note that it does not matter if x is even or odd).*

   Note that if you chose 1^p01^p *then you cannot pump it.*

6. Prove or disprove the following statements. Assume that $\Sigma = \{0,1\}$

   a. 0^n1^{2n} is regular, $n \geq 1$

.   Circle one: Prove (statement true) Disprove (statement false)

   Prove or disprove below:

   If A is a regular language, then there is a number p (the pumping length) where, if s is any string in A of length at least p, then s may be divided into 3 pieces, s = xyz, satisfying the 3 pumping lemma conditions.

   Let $S = 0^p1^{2p}$

   By condition 2, y may only contain 0's. Assuming xyz is in the language, then xyyz will not be in the language since it will yield $0^{p+1}1^{2p}$. *Hence the language is not regular. Disproved.*

b. 0^n1^{2n} is context-free, $n \geq 1$

Circle one: $\boxed{\text{Prove (statement true)}}$ Disprove (statement false)

Prove or disprove below:

Proof by construction 1:
$S \rightarrow 0R11$
$R \rightarrow 0R11 \mid \varepsilon$

Proof by construction 2:
Use a PDA and start in a state that pushes a 0 on the stack for each 0 read. Once the first 1 is seen then move into another state that pops a 0 from the stack for every second 1 read. If you run out of 1's when the stack is not empty or if you see any more 0's, then reject. Accept if you run out of input, are in the state where you just popped off a 0, and if the stack is empty.

7. Provide a Context Free grammar that generates the language 00*1*.

$S \rightarrow AB$

$A \rightarrow 0A|0$

$B \rightarrow 1B|\varepsilon$

8. Provide a context free grammar that generates $L = \{a^nb^m \colon n \neq m\}$

$S \rightarrow AS_1 \mid S_1B$ // First rule generates string with more a's; second with more b's

$S1 \rightarrow aS_1b \mid \varepsilon$ // adds equal numbers of a's and b's in proper order

$A \rightarrow aA|a$ // generates one or more a's

$B \rightarrow bB|b$ // generates one or more b's

9. Answer the following questions for the language of binary palindromes, L, where for any $w \in L$, $w=w^R$. (Note this allows for palindromes of any length, odd or even)

   a. Is L regular? No proof is required. Circle one: Yes $\boxed{\text{No}}$

   b. Is L context free? Circle one: $\boxed{\text{Yes}}$ No

   c. Provide a proof for your answer in part b

      Proof by construction:

      $S \rightarrow 0 \mid 1 \mid 0S0 \mid 1S1 \mid \varepsilon$

   d. If L is context free, then provide an English language description of the PDA that recognizes L; if it is not context free, then explain why a PDA cannot be created to recognize L.

      A PDA can recognize L. First push $ onto the stack so that you know when the stack is empty. Then go into a state that pushes every input symbol onto the stack. That state has an epsilon move out of it to another state, where this move does not alter the stack. In this other state, each input symbol is read and then that same symbol is popped off of the stack (if this does not work then the computation terminates). This state is connected by another non-deterministic move that pops off the $ on top of the stack and enter an accept state.