**CS154**

Name: ________________________ Date: ____________

*Note: The purpose of the following questions is:*

| | | |
|---|---|---|
| • *Enhance learning* | • *Summarized points* | • *Analyze abstract ideas* |

## Class 06: Elementary Questions for Reg. Lang & Pumping Lemma for Regular Languages

*Membership Questions:*

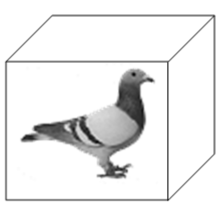
*Membership question and a method for answering it is called membership algorithm, think of it as a method for which one can write a computer program. Very little can be done with languages for which we cannot find efficient membership algorithms. It is an issue that is often difficult. For regular languages, though, it is an easy matter.*

1. Given regular language ***L*** and string ***w*** how can we check if $w \in L$ **?**
2. Given regular language ***L*** how can we check if ***L*** is empty: $(L = \phi)$?
3. Given regular language ***L*** how can we check if ***L*** is finite?
4. Given regular language $L_1$ and $L_2$ how can we check if $L_1 = L_2$?

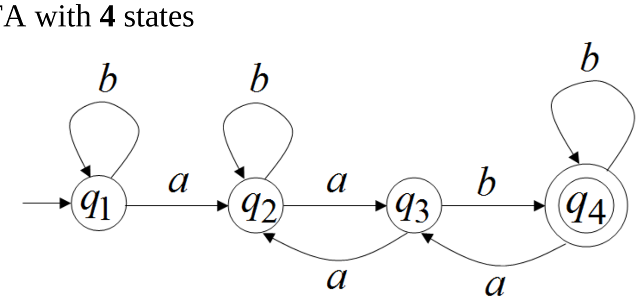
*Non-regular Languages:*

*Regular languages can be infinite, as most of our examples have demonstrated. The fact that regular languages are associated with automata that have finite memory, however, imposes some limits on the structure of a regular language. Some narrow restrictions must be obeyed if regularity is to hold. Intuition tells us that a language is regular only if, in processing any string, the information that has to be remembered at any stage is strictly limited. This is true, but has to be shown precisely to be used in any meaningful way. There are several ways in which this is can be done.*

5. How can we prove that a language ***L*** is not regular?
6. What is the pigeonhole principle? Explain how it is used by mathematicians.



7. The following DFA with **4** states



- In walks of string: ***a, aa, aab***, how many states is repeated?
- In walks of string: ***aabb, bbaa, abbabb, abbbabbabb***…, how many states is repeated?

What is your observation for this DFA?
What is your observation in general, for any DFA?

8. Considering the pigeonhole principle in the case of the DFA, what are the *pigeons* and what are the *pigeonholes*?
9. The *pigeonhole principle* is just a way of stating unambiguously, what we mean when we say that a finite automaton has a limited memory. How can we apply the *pigeonhole principle* for infinite regular language ***L?***

10. The following results known as the **pumping lemma** for regular languages, uses the *pigeonhole principle* in another form.

The Pumping Lemma:

- Given a infinite regular language $L$
- there exists an integer $m$
- for any string $w \in L$ with length $|w| \geq m$
- we can write $w = x\, y\, z$
- with $|x\, y| \leq m$ and $|y| \geq 1$
- such that: $x\, y^i\, z \in L \qquad i = 0, 1, 2, \ldots$

The proof is based on the observation that in a transition graph with ***n*** vertices, any walk of length ***n*** or longer must repeat some vertex, that is, contain a cycle. *Explain*

*11.* <u>Theorem</u>: The language $L = \{a^n b^n : n \geq 0\}$ *is not regular. Proof the theorem using the pumping lemma.*