

Pumping Lemma

Ling 106

October 8, 2003

1. What is Pumping Lemma useful for?

We know that a language is regular if we can construct a finite state automaton for it. Not all languages are regular, though. But then how would we know if a language is not regular. Could we simply conclude that a language is not regular if we cannot construct an FSA for it? Not really. It might be because we didn't think hard enough.

We need some systematic method for showing that a language is not regular, and, therefore, an FSA cannot be constructed for it. Pumping Lemma states a deep property that all regular languages share. By showing that a language does not have the property stated by the Pumping Lemma, we are guaranteed that it is not regular.

2. The idea: The Pigeon Hole Principle

Partee et al. p. 468 "Consider an infinite (regular language) L . By definition, it is accepted by some FSA M , which, again by definition, has a finite number of states. But since L is infinite, there are strings in L which are as long as we please, and certainly M contains strings with more symbols than the number of states in M . Thus, since M accepts every string in L , there must be a loop in M ..."

If p number of pigeons are placed into fewer than p holes, some holes has to have more than one pigeon in it. (Pigeon Hole Principle)

Similarly, if an FSA has n number of states, and this machine accepts strings of length n or greater, it will have to pass through at least one state more than once in order to accept such strings. That is, there will be a loop in the machine

$$q_1, q_2, q_3, \dots, q_k, \dots, q_k, \dots, q_{n-1}, q_n$$

This means that there is some substring that is read by the sequence of states:

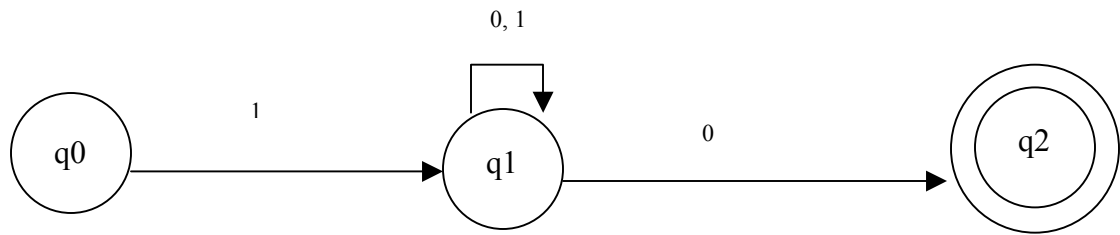
$$q_k, \dots, q_k$$

Given a string with length n or greater, which has a substring read by looping through q_k , we can construct even longer strings of the language by repeating (pumping) that substring over and over again.

3. What does Pumping Lemma say?

Theorem 1. Pumping Lemma

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



Let's assume that the pumping length is 3. Let's take some string of length 3 or longer. How about 10100010? We can break this string like this:

$$x=1, y=01, z=00010$$

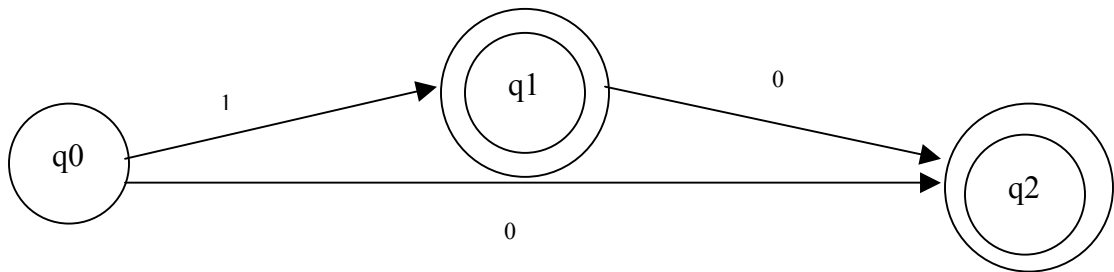
By pumping y , we get:

$xy^0z = 1-00010$, $xy^1z = 1-01-00010$, $xy^2z = 1-0101-00010$, $xy^3z = 1-010101-000010$, etc
 All of these strings begin with 1 and end with 0. So, the pumping lemma works for this language and this string.

Show that the strings 100 and 1100 in language B can also be divided in a way that complies with the Pumping Lemma.

Question:

Apply the Pumping Lemma to the language C recognized by the following FSA. Assume the pumping length is 3 and use the strings indicated below:



- a. 1001
- b. 10010
- c. 1001001
- d. 100
- e. 10

4. How to use the Pumping Lemma to prove that a language is not regular?

The pumping lemma is most useful when we want to prove that a language is not regular, We do this by using a proof by contradiction.

To prove that a given language L is not regular:

1. Assume that L is regular.
2. Use the pumping lemma to guarantee the existence of a pumping length p such that all strings of length p or greater in L can be pumped.
3. Find a string s in L that has length p or greater but that cannot be pumped.
4. Demonstrate that s cannot be pumped by considering all ways of dividing s into x , y , and z , and for each division, finding a value i where $xy^iz \notin L$
 \Rightarrow The existence of s contradicts the pumping lemma if L were regular. Hence L cannot be regular.

Example.

Let B be the language $\{0^n 1^n \mid n \geq 0\}$. Show that B is not regular, using the pumping lemma. We will do this by assuming that B is regular, and showing that contradiction follows. Therefore, the assumption we started out with must be wrong, and thus B is not regular.

Let p be the pumping length given by the pumping lemma. Choose s to be the string $0^{p-1} 1^{p-1}$

Because $s \in B$, and s has the length greater than p , the pumping lemma guarantees that we can split s into three pieces, $s = xyz$ in such a way that for any $i \geq 0$, the string $xy^i z$ is in B . We consider three cases to show that this result is impossible.

1. The string contains only of 0s. In this case, the string $xyyz$ (pumped up) has more 0s than 1s and so is not a member of B , violating condition 1 of the pumping lemma. This is a contradiction. Also, the string xz (pumped down) has more 1s than 0s and hence is not a member of B . Contradiction.
2. The string y contains only of 1s. In this case, the string $xyyz$ has more 1s than 0s and so is not a member of B . This is another contradiction.
3. The string y contains both 0s and 1s. In this case, the string $xyyz$ may have the same number of 0s and 1s, but they will be out of order with some 1s before 0s. But in our language B , all the 0s must precede the 1s. Thus $xyyz$ is not in our language. This is another contradiction.

There is no other way to split the string s , so a contradiction is unavoidable if we make the assumption that B is regular, and so B is not regular.