Module 26

- Pumping Lemma
  - A technique for proving a language L is NOT regular
  - What does the Pumping Lemma mean?
  - Proof of Pumping Lemma

Pumping Lemma

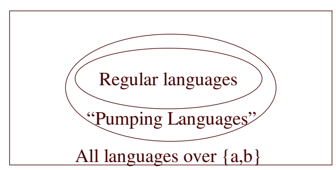
How do we use it?

Pumping Condition

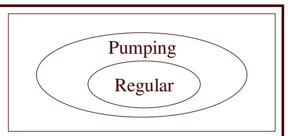
- A language L satisfies the pumping condition if:
  - there exists an integer n > 0 such that
  - for all strings x in L of length at least n
  - there exist strings u, v, w such that
    - x = uvw and
    - |uv| <= n and
    - |v| >= 1 and
    - For all k >= 0, uv^kw is in L

Pumping Lemma

- All regular languages satisfy the pumping condition



Implications *



- We can use the pumping lemma to prove a language L is not regular
  - How?
- We cannot use the pumping lemma to prove a language is regular
  - How might we try to use the pumping lemma to prove that a language L is regular and why does it fail?

Pumping Lemma

What does it mean?

Pumping Condition

- A language L satisfies the pumping condition if:
  - there exists an integer n > 0 such that
  - for all strings *x in L* of length at least n
  - there exist strings u, v, w such that
    - *x = uvw* and
    - |uv| <= n and
    - |v| >= 1 and
    - *For all k >= 0, uv^kw is in L*

7

v can be pumped

1) x in L
2) x = uvw
3) For all k >= 0, uv^kw is in L

- Let x = abcdefg be in L
- Then there exists a substring v in x such that v can be repeated (pumped) in place any number of times and the resulting string is still in L
  - uv^kw is in L for all k >= 0
- For example
  - v = cde
    - uv^0w = uw = abfg is in L
    - uv^1w = uvw = abcdefg is in L
    - uv^2w = uvvw = abcdecdefg is in L
    - uv^3w = uvvvw = abcdecdecdefg is in L
    - …

8

What the other parts mean

- A language L satisfies the pumping condition if:
  - there exists an integer n > 0 such that
    - *defer what n is till later*
  - for all strings x in L of length at least n
    - *x must be in L and have sufficient length*
  - there exist strings u, v, w such that
    - x = uvw and
    - |uv| <= n and
      - *v occurs in the first n characters of x*
    - |v| >= 1 and
      - *v is not λ*
    - For all k >= 0, uv^kw is in L

9

Examples *

- Example 1
  - Let L be the set of even length strings over {a,b}
  - Let x = abaa
  - Let n = 2
  - What are the possibilities for v?
    - abaa, abaa
    - abaa
  - Which one satisfies the pumping lemma?

10

Examples *

- Example 2
  - Let L be the set of strings over {a,b} where the number of a's mod 3 is 1
  - Let x = abbaaa
  - Let n = 3
  - What are the possibilities for v?
    - abbaaa, abbaaa, abbaaa
    - abbaaa, abbaaa
    - abbaaa
  - Which ones satisfy the pumping lemma?

11

Pumping Lemma

Proof

12

High Level Outline

- Let L be an arbitrary regular language
- Let M be an FSA such that L(M) = L
  - M exists by definition of LFSA and the fact that regular languages and LFSA are identical
- *Show that L satisfies the pumping condition*
  - Use M in this part
- Pumping Lemma follows

13

First step: n+1 prefixes of x

- Let n be the number of states in M
- Let x be an arbitrary string in L of length at least n
  - Let x_i denote the ith character of string x
- There are at least n+1 distinct prefixes of x
  - length 0: λ
  - length 1: x_1
  - length 2: x_1x_2
  - ...
  - length i: $x_1x_2 \ldots x_i$
  - ...
  - length n: $x_1x_2 \ldots x_i \ldots x_n$

14

Example

- Let n = 8
- Let x = abcdefgh
- There are 9 distinct prefixes of x
  - length 0: λ
  - length 1: a
  - length 2: ab
  - ...
  - length 8: abcdefgh

15

Second step: Pigeon-hole Principle

- As M processes string x, it processes each prefix of x
  - In particular, each prefix of x must end up in some state of M
- Situation
  - There are n+1 distinct prefixes of x
  - There are only n states in M
- Conclusion
  - At least two prefixes of x must end up in the same state of M
    - Pigeon-hole principle
  - Name these two prefixes p_1 and p_2

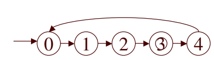
16

Third step: Forming u, v, w

- Setting:
  - Prefix p_1 has length i
  - Prefix p_2 has length j > i
    - prefix p_1 of length i: $x_1x_2 \ldots x_i$
    - prefix p_2 of length j: $x_1x_2 \ldots x_i\, x_{i+1} \ldots x_j$
- Forming u, v, w
  - Set u = p_1 = $x_1x_2 \ldots x_i$
  - Set v = $x_{i+1} \ldots x_j$
  - Set w = $x_{j+1} \ldots x_{|x|}$
  - $\underbrace{x_1x_2 \ldots x_i}_{u}\, \underbrace{x_{i+1} \ldots x_j}_{v}\, \underbrace{x_{j+1} \ldots x_{|x|}}_{w}$

17

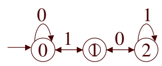
Example 1 *



- Let M be a 5-state FSA that accepts all strings over {a,b,c,…,z} whose length mod 5 = 3
- Consider x = abcdefghijklmnopqr, a string in L
- What are the two prefixes p_1 and p_2?
- What are u, v, w?

18

Example 2 *



- Let M be a 3-state FSA that accepts all strings over {0,1} whose binary value mod 3 = 1
- Consider x = 10011, a string in L
- What are the two prefixes p_1 and p_2?
- What are u, v, w?

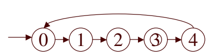
19

Fourth step: Showing u, v, w satisfy all the conditions

- |uv| <= n
  - uv = p_2
  - p_2 is one of the first n+1 prefixes of string x
- |v| >= 1
  - v consists of the characters in p_2 after p_1
  - Since p_2 and p_1 are distinct prefixes of x, v is not λ
- For all k >= 0, uv^kw in L
  - u=p_1 and uv=p_2 end up in the same state *q* of M
    - This is how we defined p_1 and p_2
  - Thus for all k >= 0, uv^k ends up in state *q*
  - The string w causes M to go from state *q* to an accepting state

20

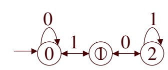
Example 1 again



- Let M be a 5-state FSA that accepts all strings over {a,b,c,…,z} whose length mod 5 = 3
- Consider x = abcdefghijklmnopqr, a string in L
- What are u, v, w?
  - u = λ
  - v = abcde
  - w = fghijklmnopqr
- |uv| = 5 <= 5
- |v| = 5 >= 1
- For all t>=0, $(abcde)^t$fghijklmnopqr is in L

21

Example 2 again



- Let M be a 3-state FSA that accepts all strings over {0,1} whose binary interpretation mod 3 = 1
- Consider x = 10011, a string in L
- What are u, v, w?
  - u = 1
  - v = 00
  - w = 11
- |uv| = 3 <= 3
- |v| = 2 >= 1
- For all k>=0, $1(00)^k11$ is in L

22

Pumping Lemma

- A language L satisfies the pumping condition if:
  - there exists an integer n > 0 such that
  - for all strings x in L of length at least n
  - there exist strings u, v, w such that
    - x = uvw and
    - |uv| <= n and
    - |v| >= 1 and
    - For all k >= 0, uv^kw is in L
- Pumping Lemma: All regular languages satisfy the pumping condition

23