CISC 4090
Theory of Computation

Non-regular languages

Professor Daniel Leeds
dleeds@fordham.edu
JMH 332

Regular languages

Definition: a language is called a regular language if some finite automaton recognizes it

What languages cannot be recognized by an FSA

Regular languages use finite memory (finite states)

Non-regular languages require infinite memory

Are the following regular?

L1 = {w | w has at least 100 1's}

L2 = {w | w has same number of 0's and 1's}

L3 = {w | w is of the form 0^n1^n, n>0}

What about this class of languages

$\Sigma = \{a, b\}$

L_n={w | w contains n b's in a row }

- L_3={abbba, aabbba,ababbbba, ...}
- L_4={babbbbab, bbbb, aaabbbbab, ...}

L_n is regular for each value of n

Regular languages can be infinite

- E.g., $a(ba)^*b$

For FSA to generate an infinite set of strings, there must be ???? between some states

5

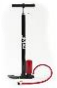
Pumping lemma

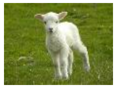
Every string in regular language L with length greater than the pumping length *p* can be "pumped"

Every string $s \in L$ ($|s|>p$) can be written as xyz where

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

If L violates pumping lemma, then it is not regular





6

Pumping lemma, continued

1. For each $i \geq 0$, $xy^iz \in L$

   There is a loop

2. $|y| > 0$

   There is a loop of letters (not of ε, which would effectively not be a loop)

3. $|xy| \leq p$

   Not allowed more states than pumping length (keep memory finite!)

7

Proof idea

If $|s| \leq p$, trivially true

If $|s|>p$, consider the states the FSA goes through

- Since there are only p states, $|s|>p$, one state must be repeated
- **Pigeonhole principle:** There must be a cycle

8

Prove B={0^n1^n} is not regular

B={01, 0011, 000111, 00001111, ….}

Proof by contradiction: assume B is regular
thus, any w ∈ B can be "pumped" if |w|>p

- First suggestion: w=0011, x=0, y=01, z=1 – counterexample
  xy^2z=001011∉ B
  Close! But maybe |0011|≤ p, how do we know this will still be a problem when string w is longer than p

Our solution: Let **w=0^p1^p** |w|>p, so **must** be "pump"-able
|xy| ≤ p so, x=0^f y=0^g, f + g ≤ p and g>0
When we pump w: xy^2z,
we get *p+g* 0's followed by *p* 1s. xy^2z ∉ B
Contradiction, pumped w ∉ B

9

Prove F={ww | w=$(0 \cup 1)^*$ } is not regular

Proof by contradiction: assume F is regular
thus, any v ∈ F can be "pumped" if |v|>p

- First suggestion: |v|=p+2 (if p even); v=ww, so |w|=$\frac{p+2}{2}$

counterexample |xy| ≤ p ≤ |w|
Say |xy|=p, v=xya where a is the first symbol of z
Pump v: xy^2z, Now xy^2a≠w, now v ≠ ww

F={11, 00, 0101, 1010, 11011101, …}

Challenge: What if we can re-group the symbols in xy^2z into a new w^{new} so $v^{pumped}=w^{new}w^{new}$? How can we guarantee this scenario won't always happen? (Intuitively, we wouldn't expect this to be a likely problem.)

10

Prove F={ww | w=$(0 \cup 1)^*$ } is not regular

Proof by contradiction: assume F is regular
thus, any v ∈ F can be "pumped" if |v|>p

- Our solution: Let w=0^p10^p1 |w|>p so must be "pump"-able
  |xy| ≤ p so, x=0^f y=0^g, f + g ≤ p and g>0
  When we pump w: xy^2z,
  we get *p+g* 0's followed by 10^p1 . xy^2z ∉ B
  Contradiction, pumped w ∉ F

F={11, 00, 0101, 1010, 11011101, …}

11

Prove E={1^{n^2}} is not regular

Proof by contradiction: assume E is regular
thus, any w ∈ E can be "pumped" if |w|>p

Our solution: Let $\mathbf{w = 1^{p^2}}$ |w|>p, so **must** be "pump"-able
|xy| ≤ p so |y| ≤ p
|xy^2z| ≤ $p^2 + p$
What's the length of the next-biggest string after |w|=p^2
|$w^{next\text{-}biggest}$| = $(p+1)^2 = p^2+2p+1$
Pumping w once gives length at most $p^2+p < p^2+2p+1$
Thus, xy^2z ∉ E
Contradiction, pumped w ∉ E

12

Common pumping proof-by-contradiction

Define a simple word w that is guaranteed to have more than p symbols, and you know the first p symbols

Show repetition of intermediate y string violates language rules

13