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

2

Are the following regular?

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

Yes: Start at q_0, For each 1 q_k -> q_{k+1} . F={q_{100}}

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

No: unknown number of states

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

No: unknown number of states

4

What about this class of languages

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

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

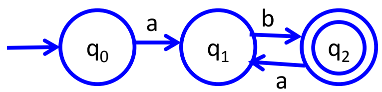
- L_3={abbba, aabbba,ababbbba, …}
- L_4={babbbbab, bbbb, aaabbbbab, …}

L_n is regular for each value of n

5

Regular languages can be infinite

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



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

6

Pumping lemma

Every string in regular language L with length greater than or equal to 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

8

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!)

9

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

10

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

B={01, 0011, 000111, 00001111, ….}

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

First suggestion: w=0011, x=0, y=01, z=1 – counterexample
xy^2z=001011$\notin B$
Close! But maybe $|0011| \leq p$, how do we know this will be problem when |w|>p

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

11

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

12

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

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

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

F={11, 00, 0101, 1010, 11011101, …}

15

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

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

Our solution: Let $\mathbf{w} = \mathbf{1^{p^2}}$ |w|>p, so **must** be "pump"-able
$|xy| \leq p$ so $|y| \leq p$
$|xy^2z| \leq p^2 + p$
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, $xy^2z \notin E$
Contradiction, pumped w $\notin E$

16

Prove $A=\{0^i1^j \mid i>j>0\}$ is not regular

Proof by contradiction: assume A is regular
thus, any $w \in A$ can be "pumped" if $|w|>p$

Our solution:

Let $\mathbf{w=0^{p+1}1^p}$ $|w|>p$, so **must** be "pump"-able

$|xy| \leq p$ so, $x=0^f$ $y=0^g$, $f+g \leq p$ and $g>0$

Let's say $xy = 0^p$ So $z=01^p$

When we pump w: xy^2z,
we get $0^f0^g0^g01^p$ -> $0^{p+g+1}1^p \in A$

Let's try pumping **down**: xy^0z,
we get xz -> 0^f01^p
Number of 0s: f+1 Number of 1s: $p=f+g\geq f+1$
$f+1\leq p$ number of 0s<number of 1 $xy^0z\notin A$

Contradiction, pumped w $\notin A$

17