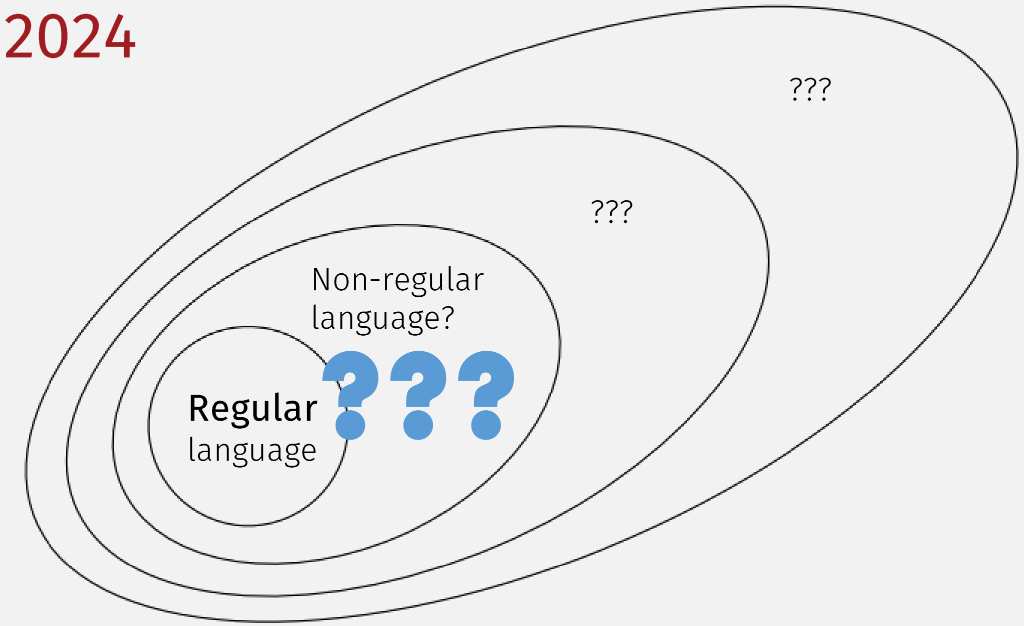
UMB CS 622

Proving Languages Non-Regular

Wednesday March 9, 2024

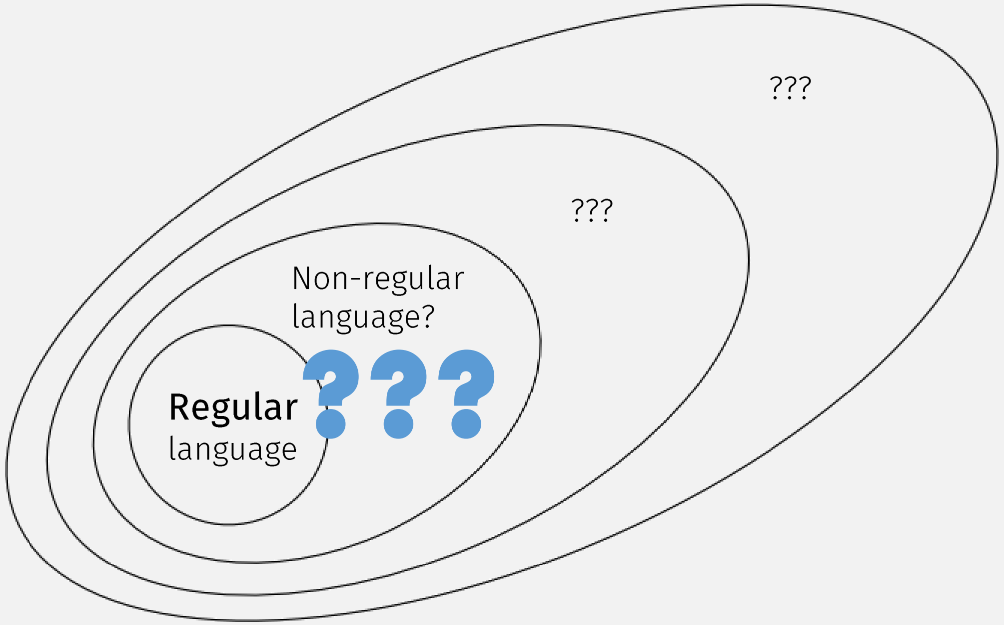


Announcements

- HW 4 out
  - Due Mon 3/18 12pm EST (noon)
  - (After spring break)

- Problem 4, Part 2c Update:
  - Prove the statement for
    - 1 base case
    - 1 recursive case

- No class next week! (Spring Break)



Last Time

Prove: Spider-Man does not exist ???

Proving something not true is **different** (and usually harder) **than proving it true**

It's sometimes possible, but **often needs** new proof techniques!

We know how to: **prove** a **language** is **regular**
Can we: **prove** a **language** is **not regular**?

Quantified Logical Statements

- **“Exists”** (Existential)
  - **“Easier”** to **prove TRUE**
    - Just need <u>one</u> **example!**

$\exists x P(x)$ is true when $P(x)$ is true for at least one value of x.

“There **exists** a **natural number** n such that, $n \cdot n = 25$”

- **“For all”** (Universal)
  - **“Harder”** to **prove TRUE**
    - Need to **prove true for** <u>all examples</u>

$\forall x P(x)$ is true when $P(x)$ is true for all values of x.

“For all natural numbers n, $2 \cdot n = n + n$”

Quantified Logical Statements in CS 622

- "Exists" (Existential)
  - "Easier" to prove TRUE → There exists one DFA that recognizes L
    - Just need one example!
  - "Harder" to prove FALSE → There are no possible DFAs that recognizes L
    - Need to prove false for all examples
- "For all" (Universal)
  - "Harder" to prove TRUE → For all regular languages L, L^* is a regular language
    - Need to prove true for all examples
  - "Easier" to prove FALSE → For all strings in a regular language …
    - Just need one (counter)example!

Language L is regular ⇓ There exists one DFA that recognizes L

Language L is not regular? ⇓ There are no possible DFAs that recognizes L ⇓ For all … ⇓ For all strings in a regular language …

Key is finding such a statement about regular languages!

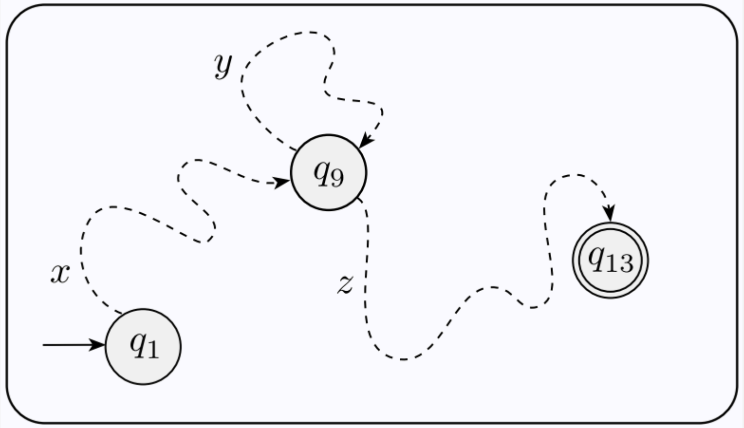
True for all regular languages!

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

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

For all (long enough) strings in the language … !

… they have this **repeatable structure** (Kleene star)



Why is this true?

Because if **you give DFA an input > # states,** then **some state repeats!**

i.e., **"long enough strings" start to show repeating pattern!**

Equivalence of Conditional Statements

- Yes or No? "If X then Y" **is equivalent to:**
  - "If Y then X" (**converse**) *Seen Previously*
    - No!
  - "If not X then not Y" (**inverse**)
    - No!
  - "If not Y then not X" (**contrapositive**)
    - Yes!

If-then statement

... then **the language is <u>not</u> regular**

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

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

Equivalent (**contrapositive**):
If **the "for all" is not true ...**
Ie **if <u>just one counterexample</u> string** doesn't have the **repeatable structure ...**

Contrapositive:
"If X then Y" is <u>equivalent</u> to "If **not** Y then **not** X"

Kinds of Mathematical Proof

- Deductive Proof
  - Logically infer conclusion from known definitions and assumptions

- Proof by induction
  - Use to prove properties of recursive definitions or functions

- Proof by contradiction
  - Proving the contrapositive

How To Do Proof By Contradiction

3 easy steps:

1. Assume: the opposite of the statement to prove

2. Show: the assumption leads to a contradiction

3. Conclude: the original statement must be true

Pumping Lemma: Non-Regularity Example

This repetition pattern cannot be expressed with a star regular expression?

Let B be the language $\{0^n 1^n | n \geq 0\}$. We use the pumping lemma to prove that B is not regular. The proof is by contradiction.

Want to prove: 0^n1^n **is not** a regular language

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

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

Reminder: Pumping lemma says: all strings $0^n1^n \geq$ length p **are splittable** into xyz where y is pumpable

So find counterexample string $\geq$ length p that is **not splittable** into xyz where y is pumpable

Proof (by contradiction):

Now we must find a contradiction …

- Assume: 0^n1^n **is** a regular language
  - So it must satisfy the pumping lemma
  - I.e., all strings $\geq$ length p are pumpable
- Counterexample = 0^p1^p

☑ In the language

☑ Greater than length p

??? Should be able to split into xyz where y is pumpable

We must show that there is no possible way to split this string to satisfy the conditions of the pumping lemma!

Want to prove: 0^n1^n **is not** a regular language

… then **not** true

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

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

Contrapositive: If **not** true …

Possible Split: y = all 0s

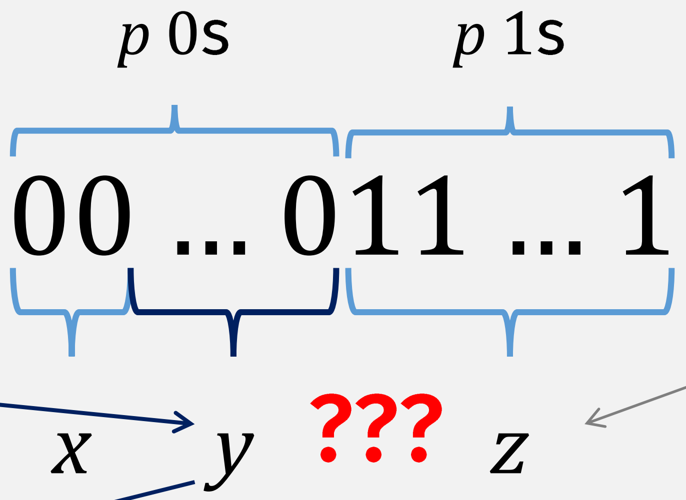
Proof (by contradiction):

- Assume: 0^n1^n **is** a regular language
  - So it must satisfy the pumping lemma
  - I.e., all strings $\geq$ length p are pumpable
- Counterexample = 0^p1^p
- Choose xyz split so y contains:
  - all 0s

Contradiction?

Not yet!

So find counterexample string $\geq$ length p that is **not splittable** into xyz where y is pumpable



BUT … pumping lemma requires **only one** pumpable splitting

This one didn't work, but proof is not done!

Is there another way to split into xyz ?

- Pumping y: produces a string with more 0s than 1s
  - … which is not in the language 0^n1^n !
  - If 0^p1^p is not pumpable? (according to pumping lemma)
  - Then 0^n1^n is a not regular language? (contrapositive)
  - This is a **contradiction** of the assumption?

Want to prove: 0^n1^n is **not** a regular language

Possible Split: y = all 1s

Proof (by contradiction):

- Assume: 0^n1^n **is** a regular language
  - So it must satisfy the pumping lemma
  - I.e., all strings ≥ length p are pumpable
- Counterexample = 0^p1^p
- Choose xyz split so y contains:
  - all 1s

p 0s $\quad$ p 1s

00 ... 011 ... 1

$x \quad y \quad z$

Is there another way to split into xyz ?

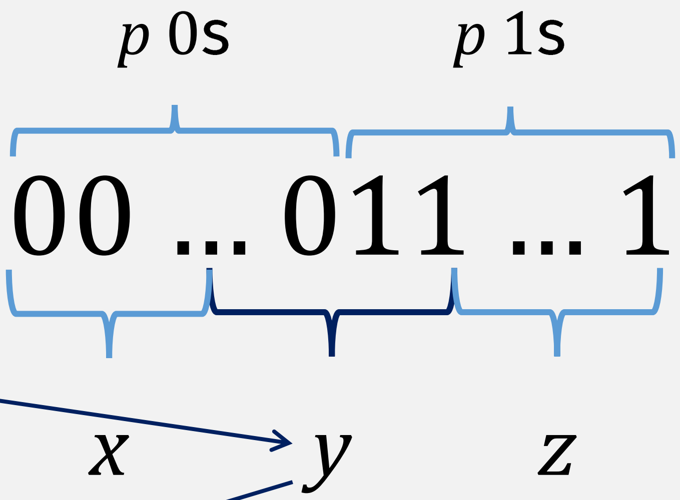
- Is this string pumpable (repeating y produces string still in 0^n1^n)?
  - No!
  - By the same reasoning as in the previous slide

Want to prove: 0^n1^n is **not** a regular language

Possible Split: y = 0s and 1s

Proof (by contradiction):

- Assume: 0^n1^n **is** a regular language
  - So it must satisfy the pumping lemma
  - I.e., all strings ≥ length p are pumpable
- Counterexample = 0^p1^p
- Choose xyz split so y contains:
  - both 0s and 1s



- Is this string pumpable (repeating y produces string still in 0^n1^n)?
  - No!
  - Pumped string will have equal 0s and 1s …
  - But they will be in the wrong order: so there is still a **contradiction**!

Did we examine every possible splitting?

Yes! QED

But maybe we did't have to …

The Pumping Lemma: Condition 3

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

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

The repeating part y …
must be in the first p characters!

i.e., "long enough strings" start to show repeating pattern!

p 0s

00 … 011 … 1

y must be in here!

(didn't have to look at other possible splittings)

The Pumping Lemma: Pumping Down

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

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

Repeating part y must be non-empty ...
but can be repeated zero times!

Example: $L = \{0^i 1^j \mid i > j\}$

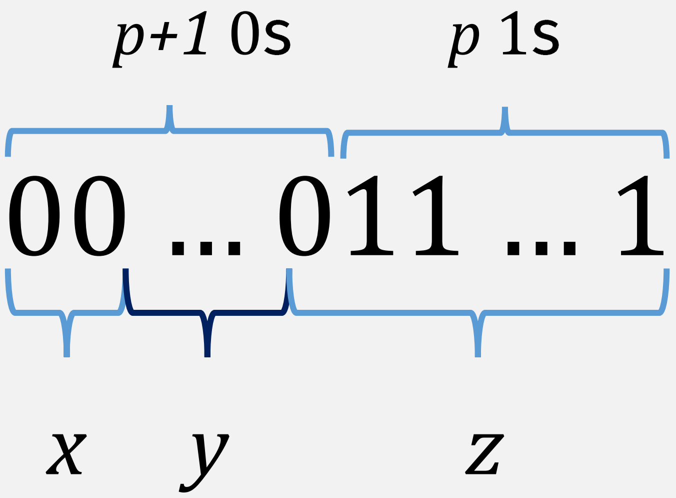
Want to prove: $L = \{0^i 1^j \mid i > j\}$ **is not** a regular language

Pumping Down

Proof (by contradiction):

contradiction

- Assume: L **is** a regular language
  - So it must satisfy the pumping lemma
  - I.e., all strings ≥ length p are pumpable
- Counterexample = $0^{p+1}1^p$
- Choose xyz split so y contains:
  - all 0s
  - (Only possibility, by condition 3)



- Repeat y zero times (**pump down**): produces string with # 0s ≤ # 1s
  - … not in the language $\{0^i 1^j \mid i > j\}$
  - So $\{0^i 1^j \mid i > j\}$ does not satisfy the pumping lemma
  - So it is a not regular language
  - This is a **contradiction** of the assumption!

Next Time (and rest of the Semester)

- If a language is not regular, then what is it?

- There are many more classes of languages!