BU CS 332 – Theory of Computation

https://forms.gle/8DBrFvVU4M7Twrsk7

Lecture 8:

- More on non-regularity
- Test 1 Review

Reading:

"Myhill-Nerode" note

Mark Bun

February 14, 2024

Last Time: Distinguishing Set Method

Definition: Strings x and y are **distinguishable** by L if there exists a "distinguishing extension" $z \in \Sigma^*$ such that exactly one of xz or yz is in L.

Definition: A set of strings S is **pairwise distinguishable** by L if every pair of distinct strings $x, y \in S$ is distinguishable by L.

Theorem: If S is pairwise distinguishable by L, then every DFA recognizing L needs at least $|S|$ states.

Corollary: If language L has an infinite pairwise distinguishable set, then L is not regular.

Reusing a Proof

Finding a distinguishing set can take some work…

Let's try to reuse that work!

How might we show that

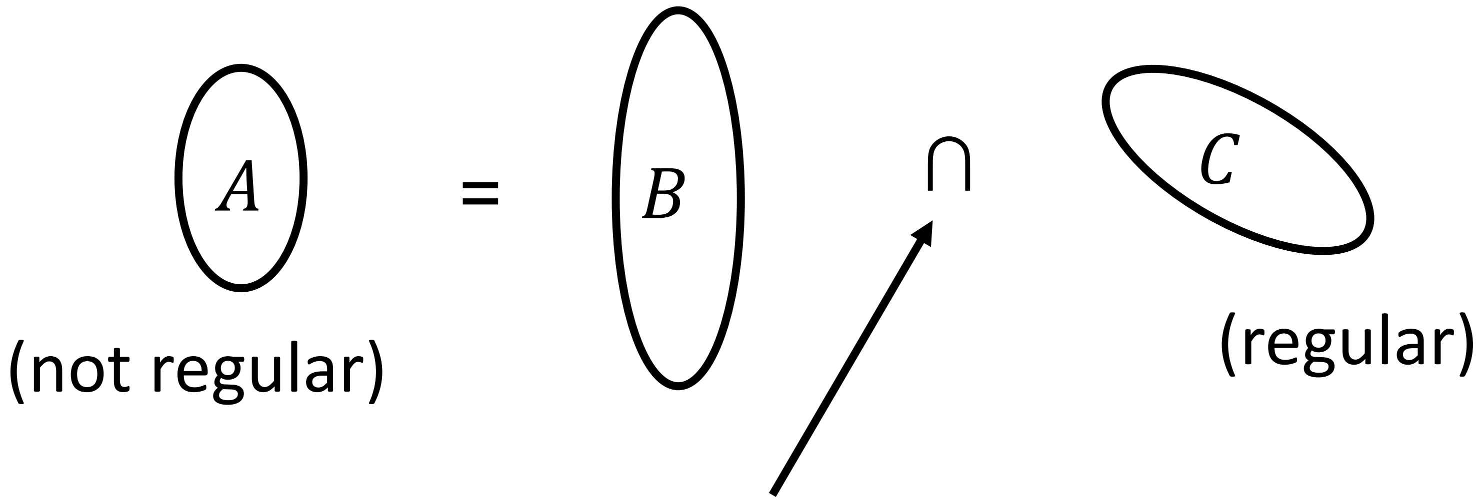
$BALANCED$ = $\{w \mid w$ has an equal # of 0s and 1s$\}$

is not regular?

$\{0^n1^n \mid n \geq 0\}$ = $BALANCED \cap \{w \mid$ all 0s in w appear before all 1s$\}$

Using Closure Properties

If A is not regular, we can show a related language B is not regular



any of $\{\circ, \cup, \cap\}$ or, for one language, $\{\neg, {}^R, {}^*\}$

<u>By contradiction:</u> If B is regular, then $B \cap C$ $(= A)$ is regular.

But A is not regular so neither is B!

Example

Prove $B = \{0^i 1^j | i \neq j\}$ is not regular using

- Nonregular language

  $A = \{0^n 1^n | n \geq 0\}$ and

- Regular language

  $C = \{w \mid \text{all 0s in } w \text{ appear before all 1s}\}$

Which of the following expresses A in terms of B and C?

a) $A = B \cap C$

b) $A = \bar{B} \cap C$

c) $A = B \cup C$

d) $A = \bar{B} \cup C$

Proof that B is nonregular

Assume for the sake of contradiction that B is regular

We know: $A = \bar{B} \cap C$

!DANGER!

Let $B = \{0^i1^j | i \neq j\}$ and write $B = A \cup C$ where

- Nonregular language

$$A = \{0^i1^j | i > j \geq 0\}$$ and

- Nonregular language

$$C = \{0^i1^j | j > i \geq 0\}$$ and

Does this let us conclude B is nonregular?

Test 1 Topics

Sets, Strings, Languages (0)

- Know the definition of a string and of a language (and the difference between them)
- Understand operations on strings: Concatenation, reverse
- Understand operations on languages: Union, intersection, concatenation, reverse, star, complement
- Know the difference between $\emptyset$ and ε

Deterministic FAs (1.1)

- Given an English or formal description of a language L, draw the state diagram of a DFA recognizing L (and vice versa)
- Know the formal definition of a DFA (A DFA is a 5 tuple…) and convert between state diagram and formal description
- Know the formal definition of how a DFA computes
- Construction for closure of regular languages under complement

Nondeterministic FAs (1.2)

- Given an English or formal description of a language L, draw the state diagram of an NFA recognizing L (and vice versa)
- Know the formal definition of an NFA
- Know the subset construction for converting an NFA to a DFA
- Proving closure properties: Know the constructions for union, concatenation, star
- Know how to prove your own closure properties

Regular Expressions (1.3)

- Given an English or formal description of a language L, construct a regex generating L (and vice versa)
- Formal definition of a regex
- Know how to convert a regex to an NFA
- Know how to convert a DFA/NFA to a regex

Limitations of DFAs (Myhill-Nerode Note)

- Understand the statements of the distinguishing set method for proving DFA size lower bounds / non-regularity
- Understand the proof of why the distinguishing set method works, and be able to use it to prove similar statements
- Know how to apply the method to specific languages
- Note: I won't ask you to show anything is non-regular, since you didn't have any homework problems on this yet

Test format

Problem 1: "Check your type checker"

E.g., Is aabba a string, language, or a regex?

How about {ab} U {aab}?

Problem 2: True/false with **justification**

Either provide a convincing explanation or a specific counterexample

Problems 3-5(?) Homework-style problems

Test tips

- You may cite without proof any result…
  - Stated in lecture
  - Stated and proved in the main body of the text (Ch. 0-1.3)
  - These include worked-out examples of state diagrams, regexes
- Not included above: homework problems, discussion problems, (solved) exercises/problems in the text

- Showing your work / explaining your answers will help us give you partial credit
- Make sure you're interpreting quantifiers (for all / there exists) correctly and in the correct order

Practice Problems

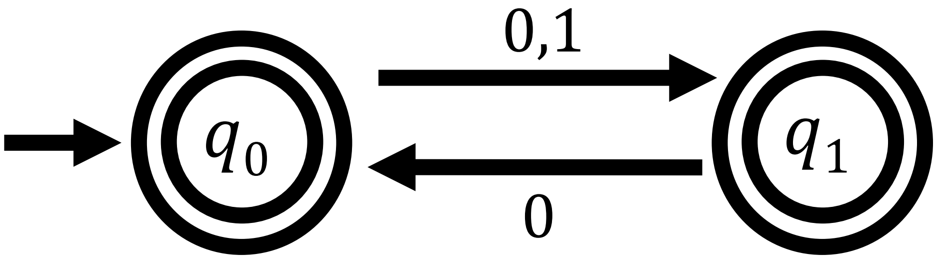
Name six operations under which the regular languages are closed

Prove or disprove: The **non-**regular languages are closed under complement

Prove or disprove: The **non-**regular languages are closed under intersection

Give the state diagram of an NFA recognizing the language $(01 \cup 10)^* \circ 1$

Give an equivalent regular expression for the following NFA



For a language L over $\{0, 1\}$, define the operation $\mathbf{split}(L) = \{x\#y \mid x, y \in L\}$. Show that the regular languages are closed under split

For a language L over alphabet Σ, define the operation $\mathbf{drop}(L) = \{xyz \mid xyz \in L, xy \in \Sigma^*, z \in \Sigma\}$. Show that the regular languages are closed under $\mathbf{drop}$.

Is the following language regular?
$\{0^n1^n \mid 0 \leq n \leq 2024\}$

How many states does a DFA recognizing $\{0^n1^n \mid 0 \leq n \leq 2024\}$ require?