San José State University
Department of Computer Science

Ahmad Yazdankhah

ahmad.yazdankhah@sjsu.edu
www.cs.sjsu.edu/~yazdankhah

Formal Languages

(Part 2)

Lecture 05

Day 05/31

CS 154

Formal Languages and Computability

Spring 2018

Agenda of Day 05

- Summary of Lecture 04
- Quiz 1
- Lecture 05: Teaching ...
  - Formal Languages (Part 2)

Summary of Lecture 04: We learned ...

Alphabets & Strings

- Alphabet is ...
  - ... a nonempty and finite set of symbols, denoted by Σ.
- String is ...
  - ... a finite sequence of symbols from the alphabet.
- Length of string w is ...
  - ... the number of symbols in the string, denoted by $|w|$.
- Empty string is ...
  - ... A string with no symbol, denoted by λ
  - $|\lambda| = 0$

Operations on Strings

- Concatenation of u and v is uv.
  - $\lambda w = w\lambda = w$ (neutral element)
- Reverse of w is denoted by w^R.
- Substring
- Prefix and Suffix
  - $w = uv$, u=prefix, v=suffix
  - λ is suffix and prefix of every string because: $w = \lambda\, w = w\, \lambda$
- Exponent operator
  - $w^n = w\, w\, w \ldots w$
  - $w\, w^n = w^n\, w = w^{n+1}$
  - $w^0 = \lambda$

Summary of Lecture 04: We learned ...

Formal Languages

- Star operator: Σ^*
  - The set of all possible strings obtained by concatenating zero or more symbols from Σ.
  - Universal set of all strings over Σ.
- Plus operator: Σ^+
  - The set of all possible strings obtained by concatenating one or more symbols from Σ.
  - $\Sigma^+ = \Sigma^* - \{\lambda\}$
  - $\Sigma^* = \Sigma^+ \cup \{\lambda\}$
- Formal language is ...
  - ... any subset of Σ^*
- Special cases:
  - $\{\ \}$ and $\{\lambda\}$

- Formal languages are sets, so, they have all sets properties.
- Formal languages can be finite or infinite.

U = All Formal Languages

Any question?

NAME	Alan M. Turing		
SUBJECT	CS 154	TEST NO.	1
DATE	02/08/2018	PERIOD	1, 2, 3

TEST RECORD	
PART 1	123
PART 2	
TOTAL	

Your list # goes here!

Quiz 1

Use Scantron

Operations on Languages

The Regular Set Operations

- Since languages are sets, we can apply all regular set operations on them.

Union

- {a, aa, ab} ∪ {a, ab, bbb, bba, b} = {a, aa, ab, bbb, bba, b}

Intersection

- {a, aa, ab} ∩ {a, ab, bbb, bba, b} = {a, ab}

Minus

- {a, aa, ab} - {a, ab, bbb, bba, b} = {aa}

Complement of Languages

Definition

- Let L be a language over a given alphabet Σ.
- Complement of L, denoted by $\overline{L}$, is defined as:

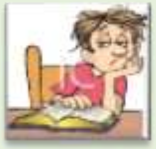
$$\overline{L} = U - L = \Sigma^* - L$$

Example 18

- Let L = {λ, b, aa, aab} over Σ = {a, b} ; $\overline{L} = ?$
- $\overline{L} = \Sigma^* - L$
- $\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}$
- $\overline{L} = \{\cancel{\lambda}, a, \cancel{b}, \cancel{aa}, ab, ba, bb, aaa, \cancel{aab}, \ldots\}$

  $= \{a, ab, ba, bb, aaa, \ldots\}$

Homework



- Given the following languages over $\Sigma = \{a, b\}$
  a. Represent L by set builder
  b. Find $\overline{L}$ and represent it by set builder

1. Set of all strings that contains at least one a
2. Set of all strings that contains more than one a
3. Set of all strings that contains exactly one a

Reverse of Languages

Definition

- Let L be a language over a given alphabet Σ.
- Reverse of L, denoted by L^R, is defined as:

$$L^R = \{w : w^R \in L\}$$

Example 19

- Let L = {b, ab, aab, abab} ; L^R = ?
- L^R = {b, ba, baa, baba}

Example 20

- Let $L = \{a^n b^n : n \geq 0\}$; L^R = ?
- $L^R = \{b^n a^n : n \geq 0\}$

Concatenation of Languages

Definition

- Let L_1 and L_2 be two languages over two alphabets Σ_1 and Σ_2.
- The concatenation of L_1 and L_2, denoted by L_1L_2, is defined as:

$$L_1L_2 = \{xy : x \in L_1, y \in L_2\} \text{ over } \Sigma = \Sigma_1 \cup \Sigma_2$$

Example 21

- Let $L_1 = \{a, ab\}$ and $L_2 = \{b, ba, baa\}$; $L_1L_2 = ?$
- $L_1L_2 = \{a, ab\} \{b, ba, baa\}$

$$= \{ab, aba, abaa, abb, abba, abbaa\}$$

Concatenation Notes

1. The concatenation of two languages looks like Cartesian product of two sets.
   - Instead of ordered-pair, we concatenate two strings.
2. $\phi\ L = L\ \phi = \phi$ (prove it!)
   - ϕ has the same role as 0 (zero) for multiplication.
3. $\{\lambda\}\ L = L\ \{\lambda\} = L$
   - $\{\lambda\}$ has the same role as 1 (one) for multiplication.
   - $\{\lambda\}$ is the neutral language for concatenation operation of languages.

Exponential Operation

Definition

- For a language L and a natural number n, L^n is defined as:

$$L^n = \underbrace{L\ L\ \ldots\ L}_{\text{n times}}$$

Example 22

Let $L = \{a, ab\}$; $L^2 = ?$; $L^3 = ?$

$L^2 = \{a, ab\}\ \{a, ab\}$

$= \{aa, aab, aba, abab\}$

$L^3 = L\ L^2 = \{a, ab\}\ \{aa, aab, aba, abab\}$

$= \{aaa, aaab, aaba, aabab, abaa, abaab, ababa, ababab\}$

- In general: $L\ L^n = L^n\ L = L^{n+1}$
- Where $n \in \mathbb{N}$ (natural numbers)

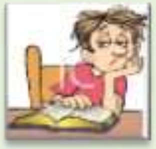
Exponential Operation

Example 23

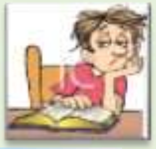
- Let $L = \{a^n b^n : n \geq 0\}$; $L^2 = ?$
- $L^2 = \{a^n b^n a^m b^m : n \geq 0, m \geq 0\}$

- Note that n and m are independent.
- For example ab aabb ($n=1$, $m=2$) belongs to L^2.

- How about $L^3 = ?$



Homework



Special cases

- $L^0 = ?$
- $L^0 = \{\lambda\}$ (prove it!)
- $\phi^0 = ?$

Optional

- For the following problems, we might use "prove by induction".
- $\{\lambda\}^n = \{\lambda\}$
- $\phi^n = ?$ for $n \geq 1$

Star-Closure Operation

Definition

- Let L be a language over a given alphabet Σ.
- Star-closure of L is defined as:

$$L^* = L^0 \cup L^1 \cup L^2 \cup \ldots$$

- Star-closure is also known as Kleene closure (pronounce it clay-knee) or Kleene star.
  - We prefer star-closure.
- The star-closure of a language L consists of all strings that can be formed by concatenating zero or more strings from L.

Star-Closure Operation

Example 24

- Let L = {a, ab} over Σ = {a, b} ; $L^* = ?$

- $L^* = L^0 \cup L^1 \cup L^2 \cup \ldots$
- $L^0 = \{\lambda\}$
- $L^1 = \{a, ab\}$
- $L^2 = \{aa, aab, aba, abab\}$
- $L^3 = \{aaa, aaab, aaba, aabab, abaa, abaab, ababa, ababab\}$
- $L^4 = \ldots$
- $L^* = \{\lambda, a, ab, aa, aab, aba, abab,$
  $aaa, aaab, aaba, aabab, abaa, abaab, ababa, ababab, \ldots\}$

Positive-Closure Operation

Definition

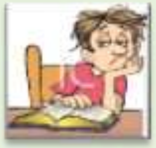
- Let L be a language over a given alphabet.
- Positive-closure of L is defined as:

$$L^{+} = L^{1} \cup L^{2} \cup \ldots$$

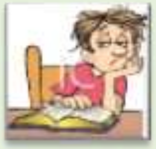
- L^{+} does not contain L^{0}. So, if we add L^{0} to L^{+}, we get L^{*}:

$$L^{*} = L^{0} \cup L^{+}$$

- Is the following statement correct?
- $L^{+} = L \; L^{*}$
- How can you support your answer?



Homework



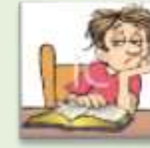
- Let L = {a, ab, bb} over Σ = {a, b}.
- $L^* = ?$
- $L^+ = ?$

- Let L = {λ, a, ab} over Σ = {a, b}.
- $L^* = ?$
- $L^+ = ?$

Special Cases

- $\phi^* = ?$
- $\{\lambda\}^* = ?$

Homework



- Enumerate at least 5 elements of the following languages:
  1. $L = \{w \in \{a, b\}^+\}$
  2. $L = \{w \in \{a, b\}^+ : |w| = 2k, K \geq 0\}$
  3. $L = \{w \in \{a, b\}^+ : |w| = 2k+1, K \geq 0\}$
  4. $L = \{1^{2k} : k \geq 1\}$ over $\Sigma = \{1\}$
  5. $L = \{w \in \{a, b\}^+ : n_a(w) = n_b(w)\}$ //number of a's = number of b's
  6. $L = \{a^n b^n c^n : n \geq 1\}$
  7. $L = \{a^n b^m c^{nm} : n, m \geq 1\}$
  8. $L = \{w\#w : w \in \{a, b\}^+\}$
  9. $L = \{w \in \{a, b\}^+ : |w| = 2k+1, K \geq 0$, w contains at least one a$\}$
  10. $L = \{ww : w \in \{a, b\}^+\}$

Surprising Languages

Surprising Languages Examples

In computer Science, all data are strings!

Example 25 : Natural Numbers

- $\mathbb{N}$ = {0, 1, ..., 123, ..., 456, ..., 5908764, ...}
- Σ = {0, 1, ..., 9}

Example 26 : Binary⁺ Numbers

- Σ = {0, 1}
- B = { 0, 1, ..., 1010, ..., 10000001, ..., 111100001, ...}

Example 27 : Unary Numbers

This is our celebrity numbers!

- Σ = {1}
- A = {1, 11, 111, 1111, 11111, ...}
- Equivalent integer numbers: 1, 2, 3, 4, 5, ...

Surprising Languages Examples

Example 28: Prime Numbers

- $\Sigma = \{0, 1, 2, ..., 9\}$
- $L = \{2, 3, 5, 7, 11, 13, 17, ...\}$

Example 29: Even and Odd Numbers

- $\Sigma = \{0, 1, 2, ..., 9\}$
- $L_1 = \{0, 2, 4, 6, 8, ...\}$
- $L_2 = \{1, 3, 5, 7, 9, ...\}$

Surprising Languages Examples

Example 30: Addition of Unary Numbers

- $\Sigma = \{1, +, =\}$
- $L = \{1^n+1^m=1^{n+m} : n \geq 1, m \geq 1\}$

- **Membership:** L contains strings such as:

  1+11=111

  11+111=11111

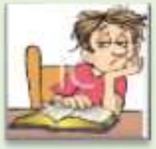
  …

- **Not Membership:** L doesn't contain strings such as:

  $1+11=1 \notin L$

  $11+111=11 \notin L$

Homework



Square of Unary Numbers

- $\Sigma = \{1, \#, =\}$
- $L = \{1^n\#{=}1^k : k = n^2, n \geq 1\}$

- **Membership:** L contains strings such as:

  ??

- **Not Membership:** L doesn't contain strings such as:

  ??

References

1. Linz, Peter, "An Introduction to Formal Languages and Automata, 5th ed.," Jones & Bartlett Learning, LLC, Canada, 2012
2. Kenneth H. Rosen, "Discrete Mathematics and Its Applications, 7th ed.," McGraw Hill, New York, United States, 2012
3. Michael Sipser, "Introduction to the Theory of Computation, 3rd ed.," CENGAGE Learning, United States, 2013
   ISBN-13: 978-1133187790