Morning

Formal Language And Automata

- Languages
- Autamata
- Computation

Language

What is a language ?

This is a sentence.

This is also a sentence.

……

So we have

{ sentence 1, sentence 2, sentence 3, …… }

the set of sentences ⇔ Language

Sentence and Alphabet

- sentence/string = sequence of symbols chosen from the alphabet Σ

  example : Mouse love rice.

- alphabet = set of symbols

  example : ASCII, 中文国标, $\Sigma = \{0,1\}$

  symbols $\Rightarrow$ sentences $\Rightarrow$ language

Rules/Grammar

- rules = by which sentence is generated

  example : rules for English

  <sentence> → <noun-phrase><predicate>

  <noun-phrase> → <article><noun>

  <predicate> → <verb>

  <article> → a|an|the

  <noun> → wolf|sheep

  <verb> → love|eat

Examples of language

example 1.1

alphabet : {0,1}

L={w|w consists of 0's and 1's,

rule

and end with 0 }

L={ 0,00,10,000,010,100,110,0000,……}

$11111100 \in L$, $1 \notin L$, $0001 \notin L$, $20 \notin L$

Examples of language

example 1.2

rule

rule

$L=\{\ 0^n1^n \mid n \geq 0\ \}$

alphabet = {0,1}

$L=\{\varepsilon, 01, 0011, 000111, 00001111, 0000011111, \ldots\ldots\}$

ε : empty string

example 1.3 The empty language

$$L = \{ \} = \phi$$

$\mathcal{L}$: contains no string

Kleene star operation

Σ^* : the set of all possible strings from alphabet Σ

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

$\Sigma^* = \{ \varepsilon,a,b,aa,ab,ba,bb,aaa,aab,aba,baa,\ldots\ldots \}$

The + operation

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

$\Sigma^+ = \Sigma^* - \{ \varepsilon \}$

$= \{a,b,aa,ab,ba,bb,aaa,aab,aba,baa,\ldots\ldots \}$

example 1.4

$$L=\{ w\in\{0, 1\}^* \mid \text{all 0's precede all 1's in } w \}$$

$$L=\{ \varepsilon, 0, 1, 00, 01, 11, 000, 001, 011, 111, \ldots\ldots \}$$

example 1.5

$L_1=\{w \in \{0,1\}^* \mid$ no prefix of w contains 1$\}$

$=\{\varepsilon, 0, 00, 000, 0000, 00000, 000000, \ldots\ldots\}$

$L_2=\{w \in \{0,1\}^* \mid$ no prefix of w starts with 1$\}$

$=\{\varepsilon, 0,00,01,000,001,010,011,0000,0001, \ldots\ldots\}$

$=\{w \in \{0,1\}^* \mid$ the first character of w is 0 $\} \cup \{\varepsilon\}$

$L_3=\{w \in \{0,1\}^* \mid$ every prefix of w starts with 1$\}$

$= \phi$

example 1.6

$L = \{ w \mid w \text{ is a sentence in English} \}$

Bolt won two gold medals in Daegu. √

The sheep eats grass . √

The grass eats sheep . ?

♣ Formal language focus on form, not meaning

String operations

$w=a_1a_2 \ldots\ldots a_n$ abba

$v=b_1b_2 \ldots\ldots b_n$ bbbaaa

Concatenation

$wv=a_1a_2 \ldots\ldots a_nb_1b_2 \ldots\ldots bn$ abbabbbaaa

Reverse

$w^R=a_na_{n-1} \ldots\ldots a_1$

Another operation

$$w^n = \underbrace{w\ w\ \ldots\ldots\ w}_{n}$$

- $w = abb \Rightarrow w^2 = abbabb,\ \ w^3 = abbabbabb$
- definition : $w^0 = \varepsilon$

The empty string

A string with no letters : ε

- $|\varepsilon| = ?$
- $w\varepsilon = ?$ $\varepsilon w = ?$

Operations on languages

- The usual set operations

$$L_1 \cup L_2 = \{ w \mid w \in L_1 \text{ or } w \in L_2 \}$$

$$L_1 \cap L_2 = \{ w \mid w \in L_1 \text{ and } w \in L_2 \}$$

$$L_1 - L_2 = \{ w \mid w \in L_1 \text{ and } w \notin L_2 \}$$

- Reverse

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

Example $L = \{ a^n b^n \mid n \geq 1 \} \Rightarrow L^R = \{ b^n a^n \mid n \geq 1 \}$

Operations on languages

- Concatenation

$$L_1L_2 = \{ wv \mid w \in L_1 \text{ and } v \in L_2 \}$$

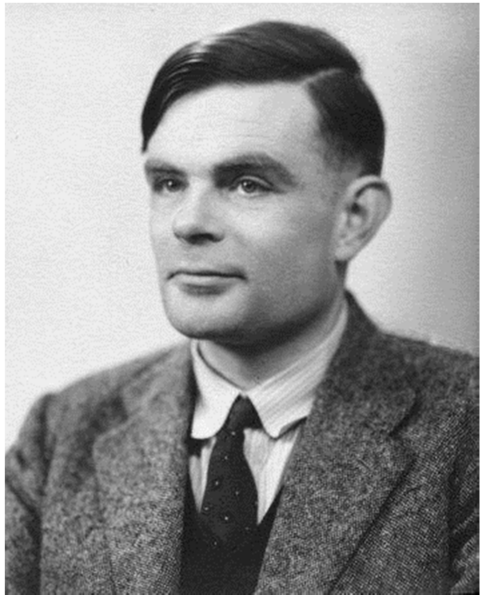
Example $\quad L = \{ a^n b^n \mid n \geq 1 \}, K = \{ b^n a^n \mid n \geq 1 \}$

$$LK = \{a^n b^n b^n a^n \mid n \geq 1 \}$$

$$LK = \{a^n b^n b^m a^m \mid n \geq 1, m \geq 1 \}$$

$$L^2 = ?$$

Automata



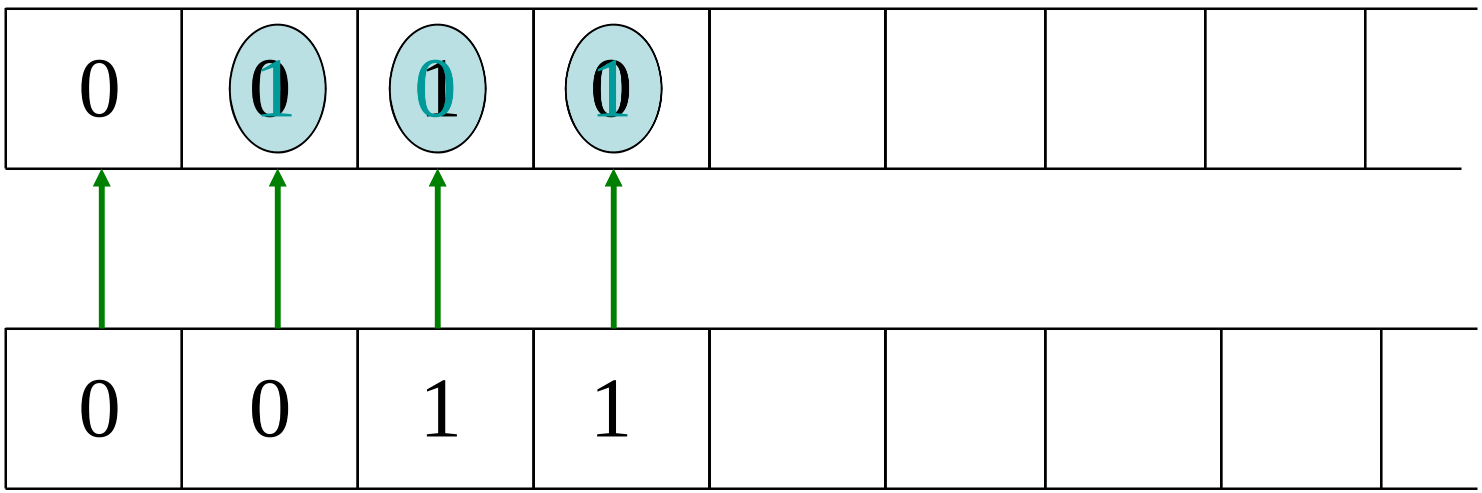
- Alan MarthisonTuring
  - On Computable Numbers With an Application to the Entscheidungs Problem
  - Turing Machine

Automata

- Finite Automata
  - Deterministic Finite Automata
  - Non-deterministic Finite Automata
- Push Down Automata

Computation

	2	0 0 1 0
+	3	+ 0 0 1 1
	5	0 1 0 1



Computation for computer

••••••

In our life

Coded

In computer

Process

0011101100001011 1100100111100
0011101100001011 11001001111000
00111011000010 11001001111

Produce

Automaton

(formal description)

Language

(set of strings)

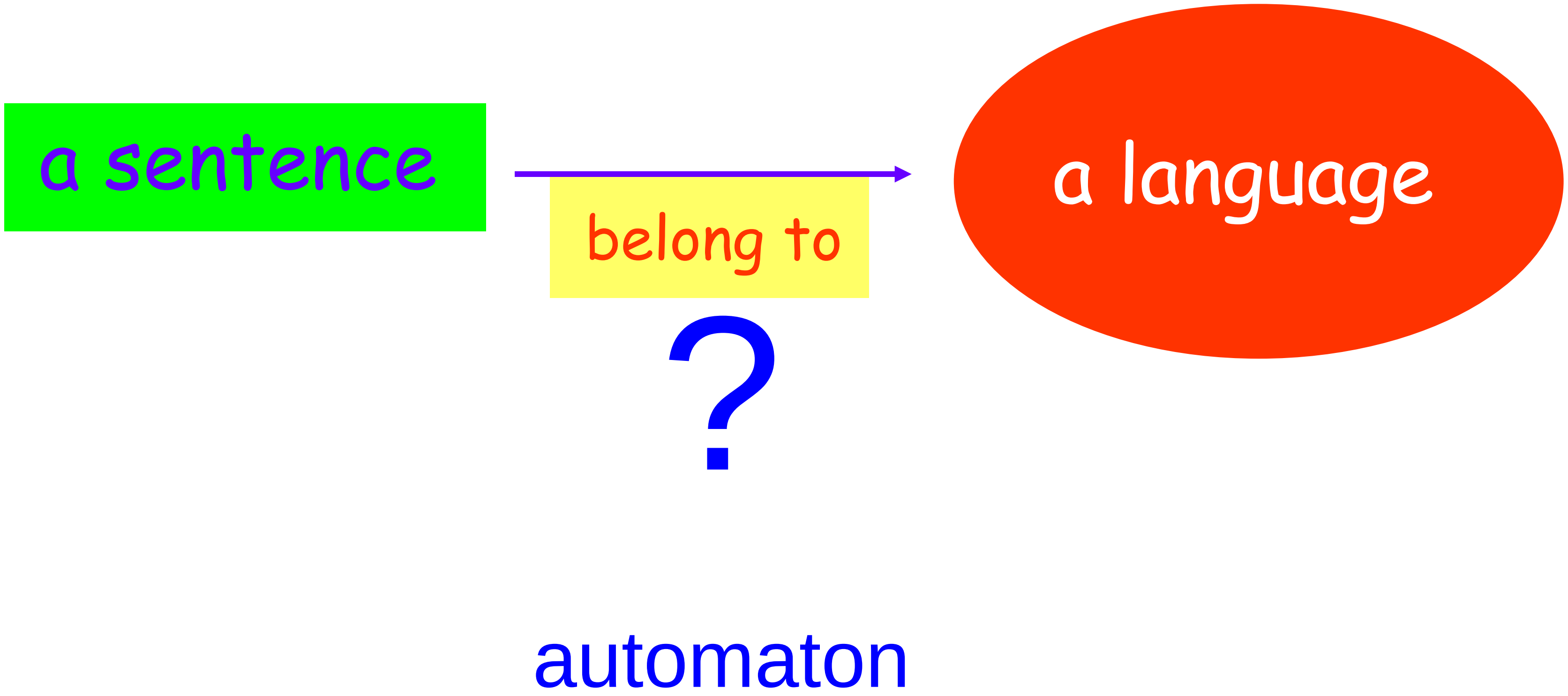
Computation

- Computable Problems
  - write a program to solve
- Intractable Problems
  - find someway to work around

Undecidable Problem

```
main ( )
{
    Int n, total, x, y, z ;
    scanf("%d", &n);
    total=3;
    while(1){
        for(x=1;x<=total-2;x++)
            for(y=1;y<=total-x-1;y++){
                z=total-x-y;
                if(exp(x,n)+exp(y,n)==exp(z,n))
                    printf("hello,world\n");
             }
        total++;
     }
}
```

Undecidable Problem



Content

Automata	Languages	Grammars
Construction	Properties	Design
Finite Automaton	Regular Language	Regular Expression
Recognize		Generate
Push Down Automaton	Context Free Language	Context Free Grammar
Turing Machine	Recursively Enumerable	(Phrase Grammar)

Text book

1. Introduction to Automata Theory , Languages , and Computation (Third Edition)

—— John E. Hopcroft
Rajeev Motwani
Jeffrey D. Ullman

Text book

2. An Introduction to Formal Languages and Automata (Third Edition)

—— Peter Linz

Goal

1. Understanding "theoretical" concepts

------ method of formal description

2. Get a sense of how to reason formally

3. Improving reading in English

Homework

- All exercises listed on cms.hit.edu.cn
- Need not submit
- Check by free talk
- One or two times of free talk

Project

- Design a DFA for the elevator
- Team member 2-3
- Presentation

Honor and Collaboration

- Collaboration is strongly encouraged
- Solutions must be written independently
- Responsible for Understanding and explaining

Exam

- Only final exam
- Open exam

  You are allowed to refer to text-book, class handouts, and notes during the exam
- Closed exam

  Nothing allowed except one pen

Grading Policy

- Homework : 20%
- Project : 20%
- Final exam : 60%

Instructor

Name → Sun Dalie | 孙大烈

Office → No. 218, Computer Building

Tel → 13936169569

E-mail → sdl@hit.edu.cn

Good good study

day day up !