计算理论——第一周

课程信息

邮箱

- maoyc@zju.edu.cn

教材

- Elements of the Theory of Computation

成绩

- 作业10%
- 期中20%
- 期末70%

Introduction

Theory of Computation(计算理论概论)

Automator Theory

- 自动机理论
- computation device(计算设备)
- language(语言)

Computability Theory

- 可计算理论

Complexity Theory

- 复杂性理论
- resource
  - time
  - space
- Complexity class
  - P and NP(non-determinist polynomial)
  - PSPACE
  - EXPTION
- 不可解决的问题
  - halting problem(停机问题)

复习离散数学的知识

- 集合
- 关系
- 函数

语言(Language)

alphabet(字母表)

Definition

- any sets being non-empty and finite
  - The elements in the set also are called symbol
  - A string over an alphabet is a finite sequence of symbols in this alphabet
  - The length of the string is the number of symbols in w, when w is an alphabet.
    - |w|
  - Let Σ be an alphabet:
    - Σ^* = {all string of Σ}
      - this is an infinite set
    - concotenation x and y
      - $x \circ y = xy$
        - 连接
    - v is a substring of w if $w = xvy$ for some $x, y \in \Sigma^*$
    - v is a prefix of w if $w = vx$ for some x in Σ^*

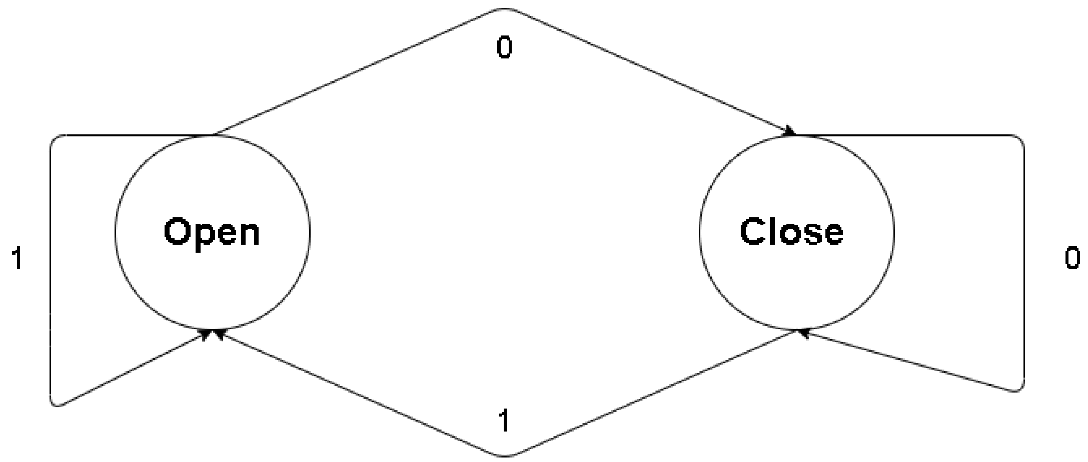
        - 从头开始的字符串子串
    - v is a sufix of w if $w = xv$ for some x in $\Sigma*$
        - 从结尾开始的字符串子串
- string exponentiation(字符乘方)
    - $w^0 = e$(e代表的是空集)
    - $\forall i \geq 0, w^{i+1} = w^i \circ w$

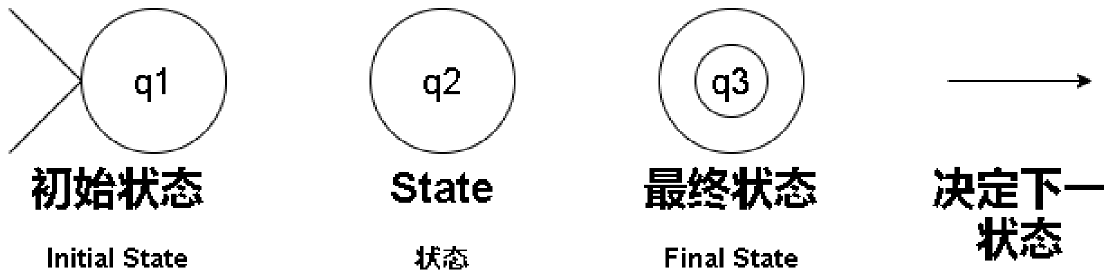
Language

Definition

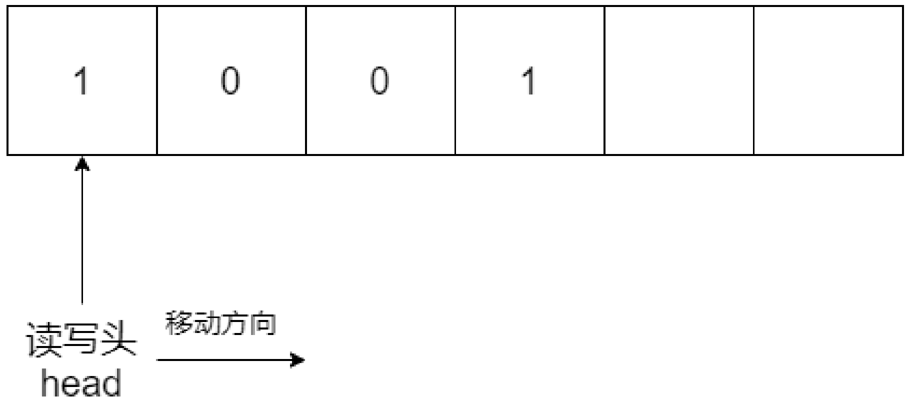
- A language over an alphabet Σ is a subset of Σ^* (finite and infinite)

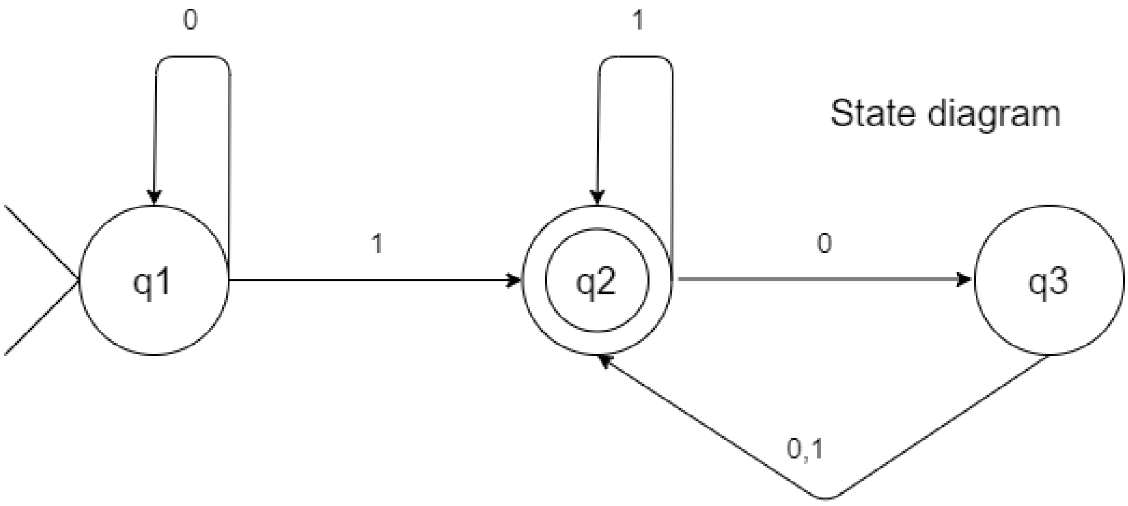
Finite Automata(有限的自动装置)

- Device with extremely limitted memory(内存机器有限的设备)
    - 自动门模型
        - 

    - 更一般的模型
        - 

    - 简单演示



- 

设备的严格定义

- K
  - a finite set of states
  - 状态集合
- Σ
  - an alphabet
- δ
  - transition function
- S
  - $S \in K$
  - initial state(unique)
- F
  - $F \subseteq K$
  - the set of final states(no unique)
- 在上面的简单演示模型中
  - $K = \{q_1, q_2, q_3\}$
  - $\Sigma = \{0, 1\}$
    - symbols read by head
  - $S = q_1$
  - $F = \{q_2\}$

- δ
  - $$K\times \Sigma \rightarrow K^*$
    - K means current state
    - Σ means symbols read by head
    - K^* means next state

$(q_1,0)$	q_1
$(q_1,1)$	q_2
...	...

Configuration

- A Configuration of $FA(F,\Sigma,\delta,S,K)$ is an element of
  - $K\times\Sigma^*$
    - K means current state
    - Σ^* means unread string

Yields in One Step(走一步)

- $(q,w)\vdash_M (q',w')$
  - if $\exist a\in\Sigma, w=aw'$ and $q'=\delta(q,a)$
- 拿上面的简单演示举例子，有:
  - $(q_1,1001)\vdash_M (q_2,001)$
  - $(q_2,001)\vdash_M (q_3,01)$
  - $(q_3,01)\vdash_M (q_2,1)$
  - $(q_2,1)\vdash_M (q_2,e)$

Yields in more than one step

- $(q,w)\vdash_M^* (q',w')$
  - if $\exist(q_1,w_1),\dots(q_i,w_i)$ and we have $(q,w)\vdash_M (q_1,w_1)\vdash\dots\vdash (q_i,w_i)\vdash\dots\vdash (q',w')$
- 拿上面的简单演示举例子，有:
  - $(q_1,1001)\vdash_M^* (q_2,e)$

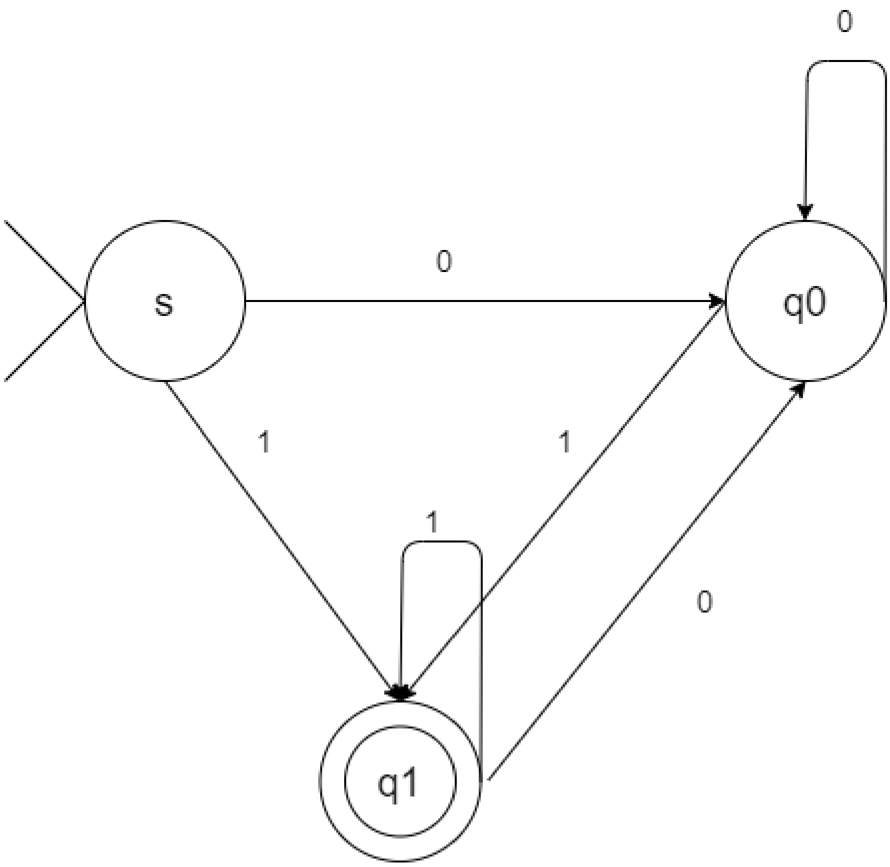
Accept

- A string $w\in\Sigma^*$ accepted by a $FA\ M=(K,\Sigma,\delta,S,F)$ if
  - $(s,w)\vdash_M^* (q,e)$ for $q\in F$
  - 即被接受的意思是，通过一系列的操作后最终能够到达终止状态
    - 如果没有能到达终止状态的话，就会被拒绝(所以上面的简单示例是会被接受的)
  - FA中的状态(Σ)不一定需要是$0,1$，在选定一个FA以后，相应的$alphabet(\Sigma)$也就被给定了
- The language accepted by a $FA\ M=(K,\Sigma,\delta,S,F)$ is the set of all strings accepted by M
  - $L(M)=\{all\ string\ accepted\ by\ M\}$
- A language is regular if it is accepted by some FA

Operation

- Suppose A, B are language over Σ
- Union
  - $A \bigcup B = \{x|x \in A \; or \; x \in B\}$
- Concotenation
  - $A \circ B = \{xy|x \in A \; and \; y \in B\}$
- (Kleene) Star
  - $A^* = \{x_1, x_2, \ldots x_k|k \geq 0 \; and \; x_i \in A\}$
    - when $k = 0, A^* = \{e\}$

Example

- Show that $L = \{w \in \{0,1\}^* : w \; ends \; with \; 1\}$ is regular
- Suppose
  - $K = S$
  - $\Sigma = \{0,1\}$
  - $F = \{q_1\}$
  - $S = S$
- Draw the diagram
  - 

- δ
  - w ends with 1
  - All satisfy
    - $(s, w) \vdash_M^* (q_1, e)$ for $q_1 \in F$

$(s, 0)$	q_0
$(q_0, 1)$	q_1
$(s, 1)$	q_1

$(s, 0)$	q_0
$(q_0, 0)$	q_0
$(q_0, 1)$	q_1
$(q_1, 1)$	**q_1**

Theory of Computation
Q & A

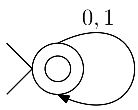
Yuchen Mao

1. *Question*: Does a language always contain the empty string e?
   Answer: No. For example, $A = \{0\}$ contains only one string 0, and it does not contain e.

2. *Question*: Can substring, prefix, suffix be empty string e?
   Answer: Yes. Indeed, by definition, e is a substring (prefix, suffix) of every string.

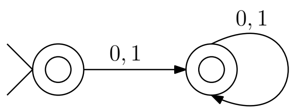
3. *Question*: Can the initial state s also be a final state?
   Answer: Yes. Think about the definition of finite automata. The definition does not prevent F from containing s.

4. *Question*: Given a finite automaton M, is the language $L(M)$ accepted by M unique?
   Answer: Yes. It is unique. $L(M)$ is defined to be the set of all strings accepted by M. In other words, $w \in L(M)$ **if and only if** w is accepted by M. For example, consider the following automaton M.

   M accepts every string it receives, so $L(M) = \Sigma^*$. Σ^* is the only language accepted by M. No other language is accepted by M.

5. *Question*: Given a regular language L, is the automaton that accepts L unique?
   Answer: No, two different automata may accept the same language. For the example, the following finite automaton is different from the one in the previous question, but it accepts Σ^*, too.

6. *Question*: Given a configuration (q, w), does (q, w) yield itself? In other words, is $(q, w) \vdash_M^* (q, w)$ true?
   Answer: Yes, $(q, w) \vdash_M^* (q, w)$ is always true.
   *Question**: But according the definition of $\vdash_M^*$ you gave in the class, this is not true.
   *Answer**: Sorry, I gave an incomplete definition in the class. The correct definition is the following

   > Given two configurations (q, w) and (q', w'), we say $(q, w) \vdash_M^* (q', w')$ if the condition I gave in the class holds or $(q, w) = (q', w')$.