

《算法设计与分析》

第九章 计算复杂性分析

马丙鹏

2023年11月27日



中国科学院大学

University of Chinese Academy of Sciences 1

第九章 计算复杂性分析

- 9.1 复杂性分类
- 9.2 P 类问题和 NP 类问题
- 9.3 NP 完全问题



9.1 复杂性分类

■ 1. 什么是计算

- 计算机科学的研究目标——用计算机来求解人类所面临的各种问题，问题本身的内在复杂性决定了求解这个问题的算法的**计算复杂性**（computing complexity）。
- 如何判定一个问题的内在复杂性？
- 如何区分一个问题是“**易解**”的还是“**难解**”的？
 - 许多情况下，要确定一个问题的内在复杂性是很困难的，人们对**许多问题至今无法确切地了解其内在的计算复杂性**。
 - **NP完全理论**从计算复杂性的角度研究**问题的分类**以及问题之间的关系，从而为算法设计提供指导。



9.1 复杂性分类

■ 1. 什么是计算

□从字源上考察：计算的原始含义是利用计算工具进行计数。

➤计：从言从十，有数数或计数的含义；

➤算：从竹从具，指计算工具。

□直观的计算：

➤数的加减乘除；函数的微分、积分；微分方程的求解；定理的证明推导等等

➤从符号串“ $12+3$ ”变换成符号串“ 15 ”——加法；

➤符号串“ x^2 ”变换成符号串“ $2x$ ”——微分；

➤ f 表示一组公理和推导规则， g 是一个定理，从 f 到 g 的一系列变换——定理 g 的证明；



9.1 复杂性分类

■ 1. 什么是计算

□ 直观的计算：

- f 代表一个英文句子， g 为含义相同的中文句子，从 f 到 g 的变换——英文翻译成中文；
- f 代表一个原始音频文件， g 为对应的 mp3 文件，从 f 到 g 的变换——数据压缩；从 g 到 f 的变换——解压缩。

□ 计算的实质：

- 从一个符号串 f （输入）得出另一个符号串 g （输出）

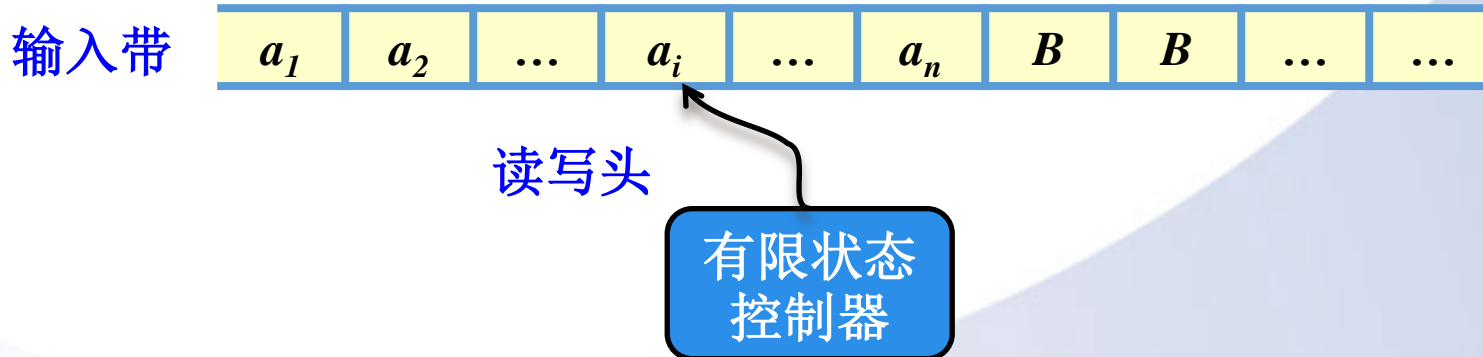


9.1 复杂性分类

■ 1. 什么是计算

□ 计算的定义：

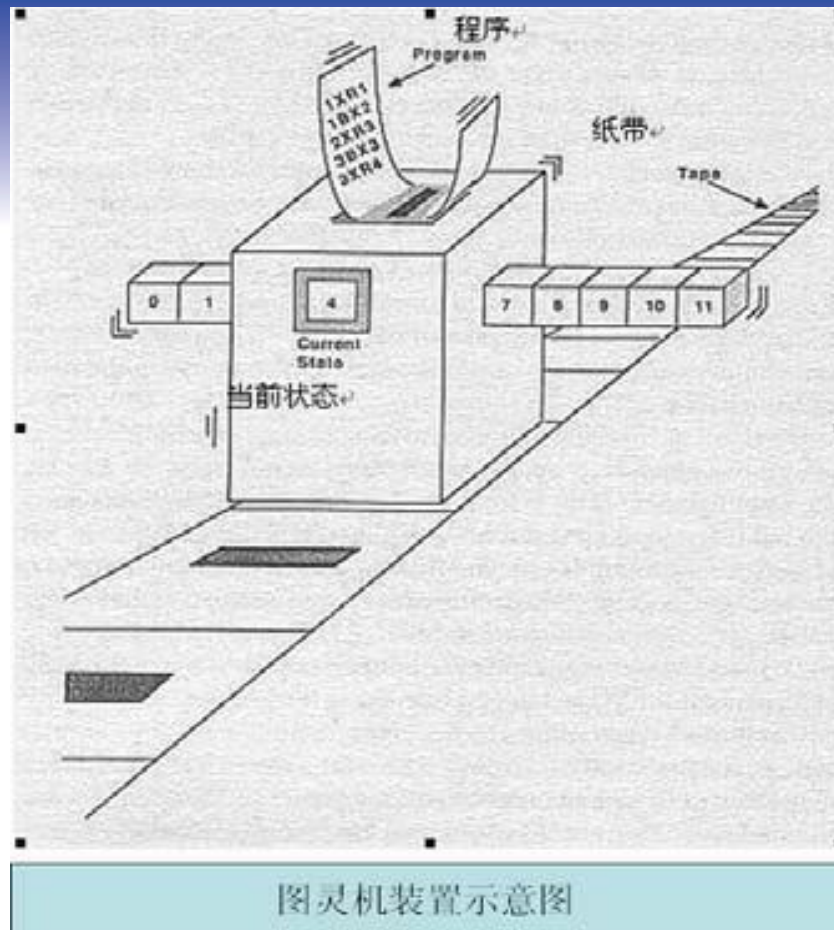
- 所谓计算就是计算者（人或机器）对一条可以无限延长的工作带上的符号串执行指令，一步一步地改变工作带上的符号串，经过有限步骤，最后得到一个满足预先规定的符号串的变换过程。



9.1 复杂性分类

■ 2. 图灵机

□ 图灵机的结构:



输入带



读写头

有限状态
控制器



中国科学院大学

University of Chinese Academy of Sciences 7

9.1 复杂性分类

■ 2. 图灵机

□ 图灵机由以下几个部分组成：

① 一条无限长的纸带 TAPE。

➤ 纸带被划分为一个接一个的小格子，
每个格子上包含一个来自**有限字母表**的符号，
字母表中**有一个特殊的符号表示空白**。
纸带上的格子从左到右依此被编号为0, 1, 2,...，纸带的右端可以无限伸展。

② 一个读写头 HEAD。

➤ 该读写头可以在纸带上**左右移动**，它能读出当前所指的格子上的符号，并能改变当前格子上的符号。



9.1 复杂性分类

■ 2. 图灵机

□ 图灵机由以下几个部分组成：

③ 一套控制规则 TABLE。

➤ 它根据当前机器所处的状态以及当前读写头所指的格子上的符号来确定读写头下一步的动作，并改变状态寄存器的值，令机器进入一个新的状态。

④ 一个状态寄存器。

➤ 它用来保存图灵机当前所处的状态。
图灵机的所有可能状态的数目是有限的，并且有一个特殊的状态，称为停机状态。



9.1 复杂性分类

■ 2. 图灵机

□ 图灵机的工作过程:

- 计算开始时，将输入符号串放在工作带上，每个单元放一个输入符号，其余单元都是空白符，控制器处于初始状态，
- 读写头扫描工作带上的第一个符号，控制器决定下一步的动作。
- 如果对于当前的状态和所扫描的符号，没有下一步的动作，则图灵机就停止计算，处于终止状态。



9.1 复杂性分类

■ 2. 图灵机

□ 图灵机的工作过程：

- 在计算的每一步，控制器处于某个状态，读写头扫描工作带的某个单元，控制器根据当前的状态和被扫描单元的内容，决定下一步的执行动作：
 - ① 把当前单元的内容改写成另一个符号；
 - ② 使读写头停止不动、向左或向右移动一个单元；
 - ③ 使控制器转移到某一个状态。



9.1 复杂性分类

■ 2. 图灵机

□ 图灵机的形式化描述:

- 图灵机是一个五元组 $(K, \Sigma, \delta, s, H)$, 其中:
- K 是有穷个状态的集合;
- Σ 是字母表, 即符号的集合;
- δ 是转移函数, 即控制器的规则集合。
- $s \in K$ 是初始状态;
- $H \in K$ 是停机状态的集合, 当控制器内部状态为停机状态时图灵机结束计算;



9.1 复杂性分类

■ 2. 图灵机

□ 图灵机的形式化描述:

- 写在带子上的符号为一个有穷字母表: $\{S_0, S_1, S_2, S_p\}$
- 一个给定机器的程序认为是机器内的五元组 $(q_i S_j S_k R q_l \text{ 或 } q_i S_j S_k L q_l \text{ 或 } q_i S_j S_k N q_l)$ 形式的指令集
 - ✓ q_i 表示机器目前所处的状态
 - ✓ S_j 表示机器从方格中读入的符号
 - ✓ S_k 表示机器用来代替 S_j 写入方格中的符号
 - ✓ R, L, N 分别表示向右移一格、向左移一格、不移动
 - ✓ q_l 表示下一步机器的状态



9.1 复杂性分类

■ 2. 图灵机

□ 图灵机的工作原理：

- 机器从给定带子上的某起始点出发，
- 根据其初始状态及机内五元组决定其动作，经过有限步骤机器停止时，
- 带子上的信息即为机器计算的结果。



9.1 复杂性分类

■ 2. 图灵机

□ 例: 读写头

代表空白

字母表: $\{1, b\}$

机器状态: $\{q1, q2, q3\}$

带



程序

指令

机器状态

控制器

当前读到的字符

下一机器状态

当前写入的字符

读写头的动作

R:右移 L:左移 H:不动

$(q1, 1, 1, R, q1)$
$(q1, b, 1, R, q2)$
$(q2, 1, 1, R, q2)$
$(q2, b, b, L, q3)$
$(q3, 1, b, H, q3)$
$(q3, b, b, H, q3)$



中国科学院大学

University of Chinese Academy of Sciences 15

9.1 复杂性分类

■ 2. 图灵机

□ 例1: 读写头

代表空白

字母表: $\{1, b\}$

机器状态: $\{q1, q2, q3\}$

带



程序

控制器

$(q1, 1 \ 1 \ R \ q1)$
$(q1, b \ 1 \ R \ q2)$
$(q2, 1 \ 1 \ R \ q2)$
$(q2, b \ b \ L \ q3)$
$(q3, 1 \ b \ H \ q3)$
$(q3, b \ b \ H \ q3)$

指令

指令各部分的合作:

- 1) 在当前机器状态下
- 2) 判断读入的符号
- 3) 写一个符号
- 4) 控制读写头动作
- 5) 设置下一机器状态



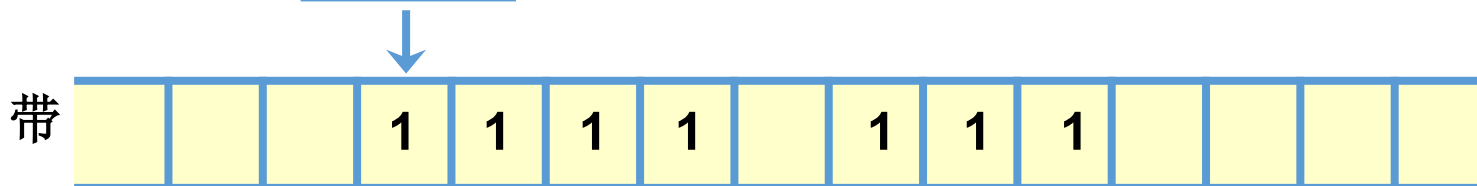
中国科学院大学

University of Chinese Academy of Sciences 16

9.1 复杂性分类

■ 2. 图灵机

□ 例1: 读写头



当前机器状态: $q1$

控
制
器

程序

$(q1, 1 \ 1 \ R \ q1)$
$(q1, b \ 1 \ R \ q2)$
$(q2, 1 \ 1 \ R \ q2)$
$(q2, b \ b \ L \ q3)$
$(q3, 1 \ b \ H \ q3)$
$(q3, b \ b \ H \ q3)$

读写头扫描到符号1,
则继续往右走



中国科学院大学

University of Chinese Academy of Sciences 17

9.1 复杂性分类

■ 2. 图灵机

□ 例1:

读写头



带



当前机器状态: $q1$

程序

控
制
器

$(q1, 1 \ 1 \ R \ q1)$

$(q1, b \ 1 \ R \ q2)$

$(q2, 1 \ 1 \ R \ q2)$

$(q2, b \ b \ L \ q3)$

$(q3, 1 \ b \ H \ q3)$

$(q3, b \ b \ H \ q3)$

读写头扫描到符号1,
则继续往右走



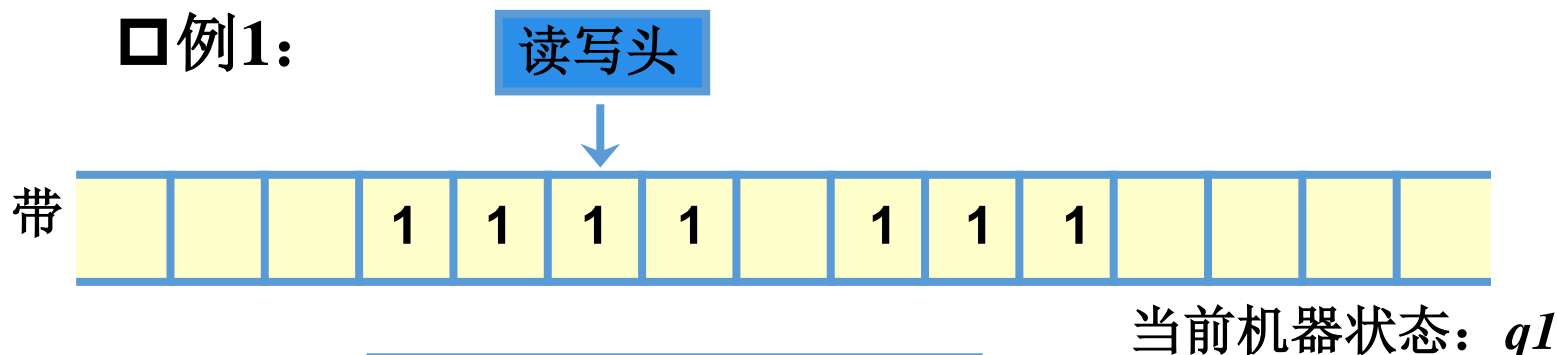
中国科学院大学

University of Chinese Academy of Sciences 18

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



控制器

程序

$(q1, 1 \ 1 \ R \ q1)$
$(q1, b \ 1 \ R \ q2)$
$(q2, 1 \ 1 \ R \ q2)$
$(q2, b \ b \ L \ q3)$
$(q3, 1 \ b \ H \ q3)$
$(q3, b \ b \ H \ q3)$

读写头扫描到符号1,
则继续往右走



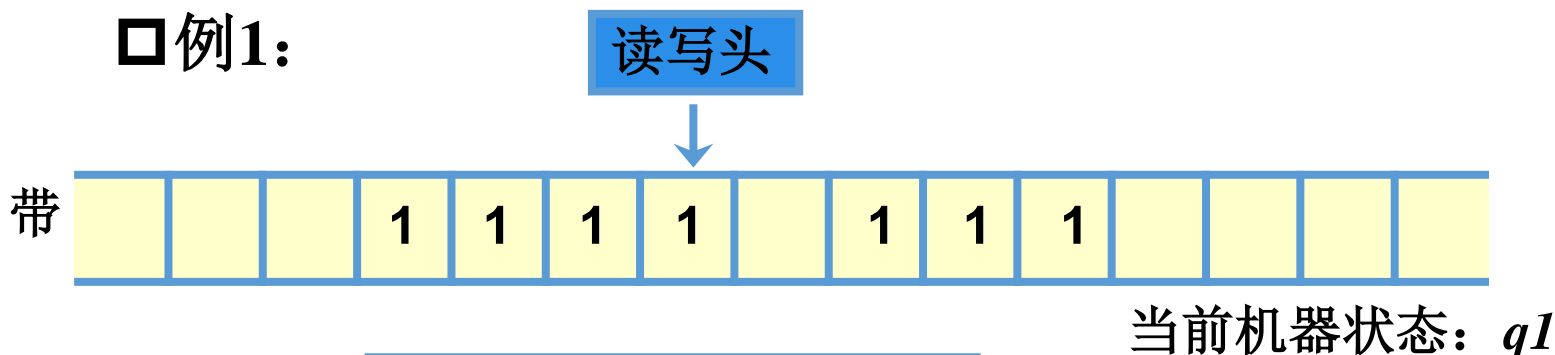
中国科学院大学

University of Chinese Academy of Sciences 19

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



控制器

程序

$(q1, 1 \ 1 \ R \ q1)$
$(q1, b \ 1 \ R \ q2)$
$(q2, 1 \ 1 \ R \ q2)$
$(q2, b \ b \ L \ q3)$
$(q3, 1 \ b \ H \ q3)$
$(q3, b \ b \ H \ q3)$

读写头扫描到符号1,
则继续往右走



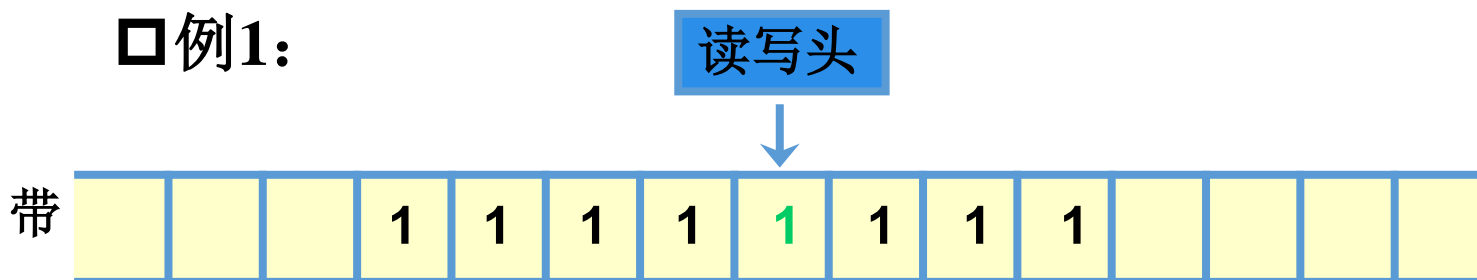
中国科学院大学

University of Chinese Academy of Sciences 20

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



当前机器状态: $q1$

控制器

程序

$(q1, 1 \ 1 \ R \ q1)$

$(q1, b \ 1 \ R \ q2)$

$(q2, 1 \ 1 \ R \ q2)$

$(q2, b \ b \ L \ q3)$

$(q3, 1 \ b \ H \ q3)$

$(q3, b \ b \ H \ q3)$

读写头扫描到空格，
写入字符1，则继续
往右走，状态变为 $q2$



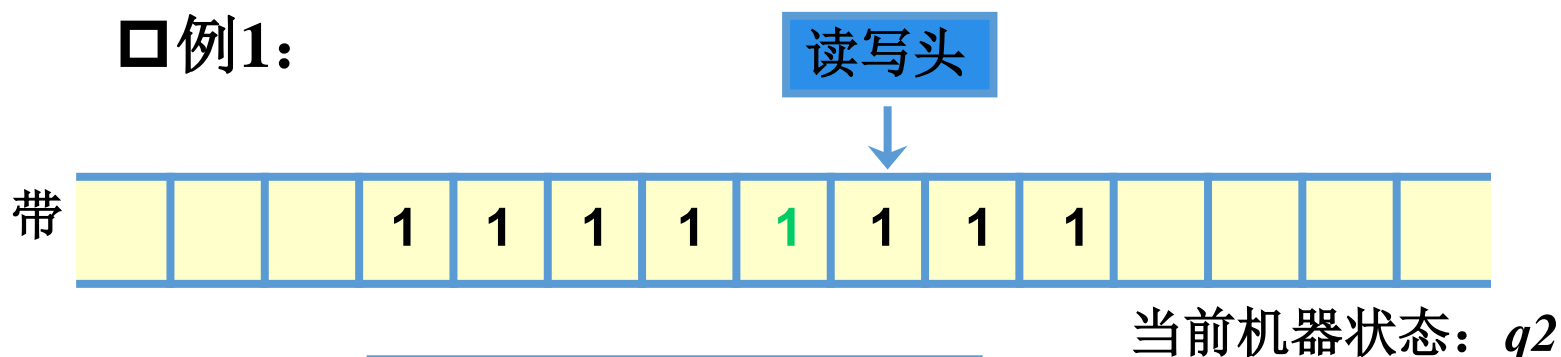
中国科学院大学

University of Chinese Academy of Sciences 21

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



控
制
器

程序

$(q_1, 1 \ 1 \ R \ q_1)$
$(q_1, b \ 1 \ R \ q_2)$
$(q_2, 1 \ 1 \ R \ q_2)$
$(q_2, b \ b \ L \ q_3)$
$(q_3, 1 \ b \ H \ q_3)$
$(q_3, b \ b \ H \ q_3)$

读写头扫描到符号 1 ,
则继续往右走



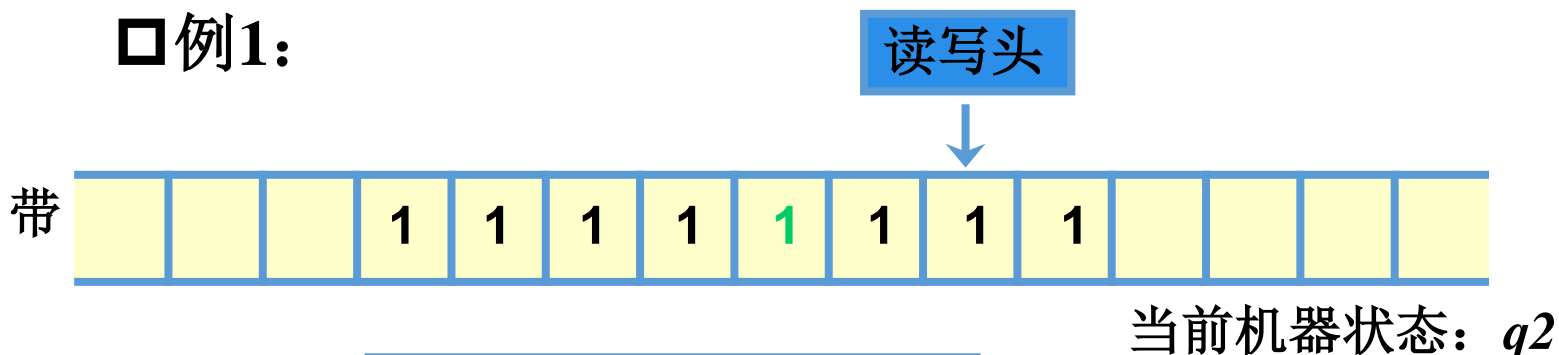
中国科学院大学

University of Chinese Academy of Sciences 22

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



控制器

程序

$(q_1, 1 \ 1 \ R \ q_1)$
$(q_1, b \ 1 \ R \ q_2)$
$(q_2, 1 \ 1 \ R \ q_2)$
$(q_2, b \ b \ L \ q_3)$
$(q_3, 1 \ b \ H \ q_3)$
$(q_3, b \ b \ H \ q_3)$

读写头扫描到符号 1 ,
则继续往右走



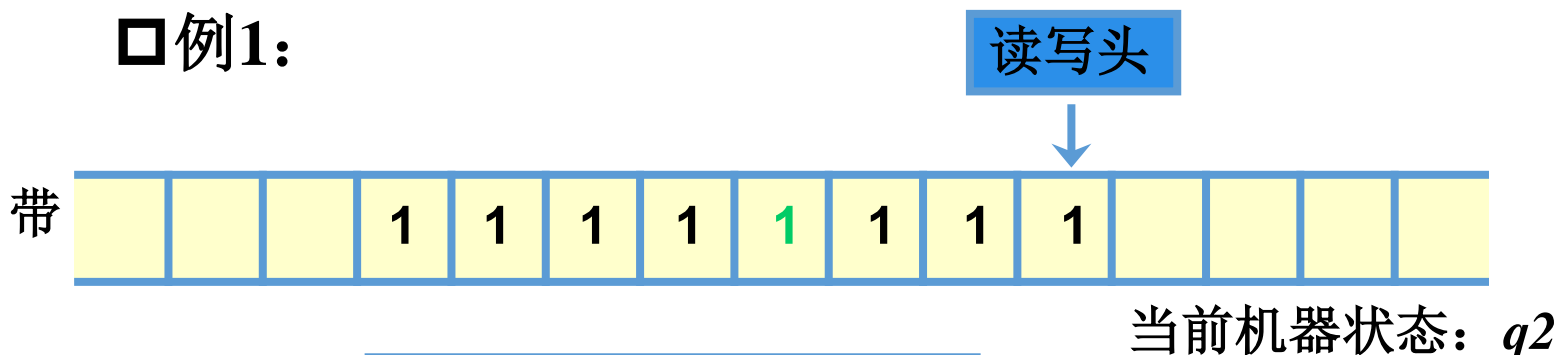
中国科学院大学

University of Chinese Academy of Sciences 23

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



控
制
器

程序

$(q_1, 1 \ 1 \ R \ q_1)$
$(q_1, b \ 1 \ R \ q_2)$
$(q_2, 1 \ 1 \ R \ q_2)$
$(q_2, b \ b \ L \ q_3)$
$(q_3, 1 \ b \ H \ q_3)$
$(q_3, b \ b \ H \ q_3)$

读写头扫描到符号 1 ,
则继续往右走



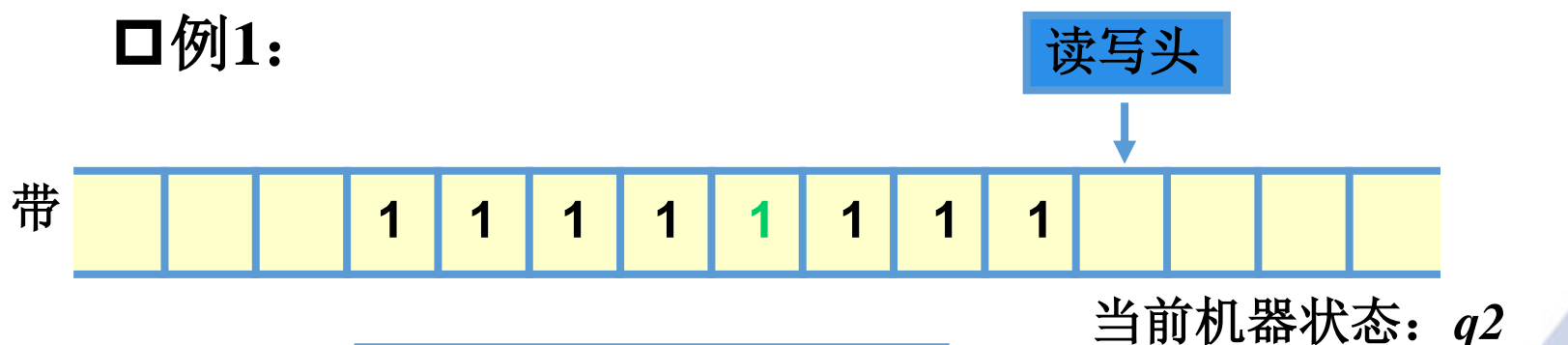
中国科学院大学

University of Chinese Academy of Sciences 24

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



控制器

程序

$(q_1, 1 \ 1 \ R \ q_1)$
$(q_1, b \ 1 \ R \ q_2)$
$(q_2, 1 \ 1 \ R \ q_2)$
$(q_2, b \ b \ L \ q_3)$
$(q_3, 1 \ b \ H \ q_3)$
$(q_3, b \ b \ H \ q_3)$

读写头扫描到空格，
则往左走，状态变为
 q_3



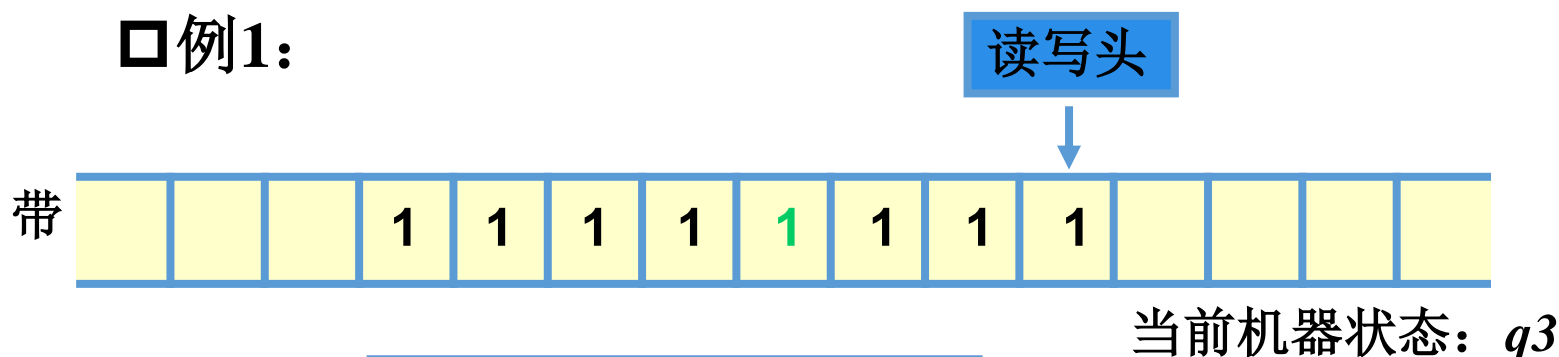
中国科学院大学

University of Chinese Academy of Sciences 25

9.1 复杂性分类

■ 2. 图灵机

□ 例1:



控制器

程序

$(q_1, 1 \ 1 \ R \ q_1)$
$(q_1, b \ 1 \ R \ q_2)$
$(q_2, 1 \ 1 \ R \ q_2)$
$(q_2, b \ b \ L \ q_3)$
$(q_3, 1 \ b \ H \ q_3)$
$(q_3, b \ b \ H \ q_3)$

读写头扫描到符号 1 ,
写入字符空格。停止



中国科学院大学

University of Chinese Academy of Sciences 26

9.1 复杂性分类

■ 2. 图灵机

□例1：这个例子中，是在用图灵机进行什么计算？

开始时：

1 1 1 1 1 1 1
 └───┘ └───┘
 x 个 1 y 个 1

结束时：

1 1 1 1 1 1 1
 └──────────┘
 x + y 个 1

□这是在任意两个大于 0 的整数的相加。



9.1 复杂性分类

■ 2. 图灵机

□例2：设b表示空格，q1表示机器的初始状态，q4表示机器的结束状态，如果带子上的输入信息是10100010，读入头对准最右边第一个为0的方格，状态为初始状态q1。按照以下规则执行之后，输出正确的计算结果。

q1 0 1 L q2

q1 1 0 L q3

q1 b b N q4

q2 0 0 L q2

q2 1 1 L q2

q2 b b N q4

q3 0 1 L q2

q3 1 0 L q3

q3 b b N q4



9.1 复杂性分类

■ 2. 图灵机

q1 0 1 L q2
 q2 0 0 L q2
 q2 1 1 L q2
 q2 b b N q4

	1	0	1	0	0	0	1	0		q ₁		q ₁ 0 1 L q ₂
	1	0	1	0	0	0	1	1		q ₂		q ₂ 1 1 L q ₂
	1	0	1	0	0	0	1	1		q ₂		q ₂ 0 0 L q ₂
	1	0	1	0	0	0	1	1		q ₂		q ₂ 0 0 L q ₂
	1	0	1	0	0	0	1	1		q ₂		q ₂ 0 0 L q ₂
	1	0	1	0	0	0	1	1		q ₂		q ₂ 1 1 L q ₂
	1	0	1	0	0	0	1	1		q ₂		q ₂ 0 0 L q ₂
	1	0	1	0	0	0	1	1		q ₂		q ₂ 1 1 L q ₂
b	1	0	1	0	0	0	1	1		q ₂		q ₂ b b N q ₄
b	1	0	1	0	0	0	1	1		q ₄		

$$S(x) = x + 1$$



中国科学院大学

University of Chinese Academy of Sciences 29

9.1 复杂性分类

■ 2. 图灵机

□例3：设计一个图灵机M，它接受的语言 $L=\{0^n1^n \mid n>1\}$ ，设计该图灵机。

- 解：假设输入串 w 为 $00...011...1BB...$ ，设计出来的图灵机的主要功能是检查0的个数和1的个数是否相等。
- 使读写头往返移动，每往返移动一次，就成对地输入符号串 w 左端的一个0和右端的一个1做标记。

0	0	1	1	B	...
---	---	---	---	---	-----



9.1 复杂性分类

■ 2. 图灵机

□例3：设计一个图灵机M，它接受的语言 $L=\{0^n1^n \mid n>1\}$ ，设计该图灵机。

- 如果恰好把输入串的全部符号都做了标记，说明左边的符号0与右边的符号1的个数相等，则w属于L；
- 否则，或者左边的0已全部标记，右边还有若干个1没有标记，或者右边的1已全部标记，左边还有若干个0没有标记，这说明左边符号0与右边符号1个数不等，或者0与1交替出现，则w不属于L。



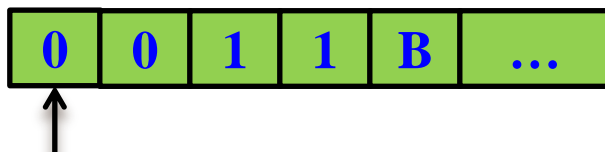
9.1 复杂性分类

■ 2. 图灵机

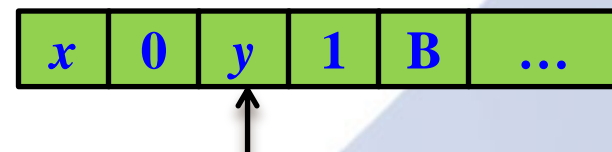
□ 例3：设计一个图灵机M，它接受的语言 $L=\{0^n1^n \mid n>1\}$ ，设计该图灵机。

➤ 例如，识别输入串 $w_1=0011$ 的过程如下：

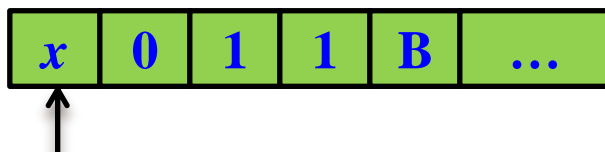
初始状态



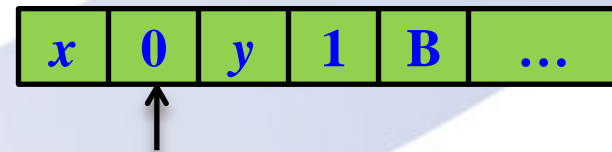
第3步：遇到1，将1改为y并左移



第1步：遇到0，将0改为x并右移



第4步：遇到0，左移



第2步：遇到0，右移

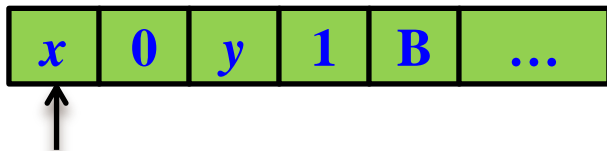


中国科学院大学

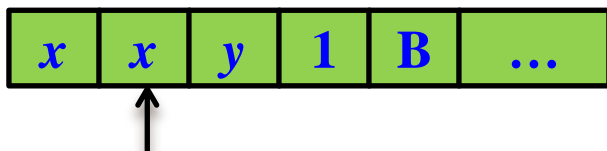
University of Chinese Academy of Sciences 32



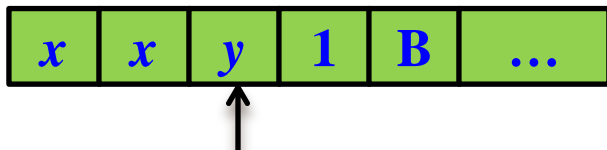
第5步：遇到 x ，右移



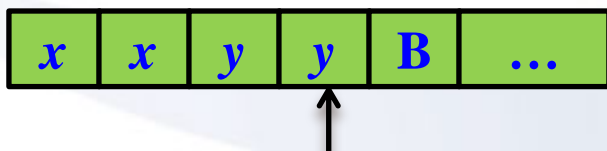
第6步：遇到0，将0改为 x 并右移



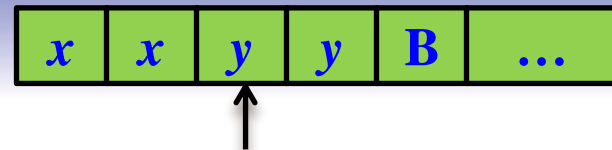
第7步：遇到 y ，右移



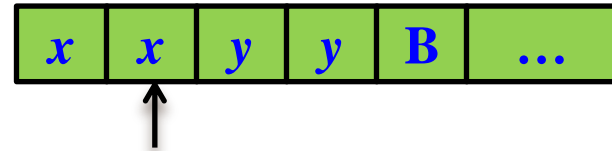
第8步：遇到1，将1改为 y 并左移



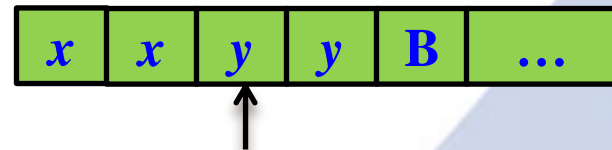
第9步：遇到 y ，左移



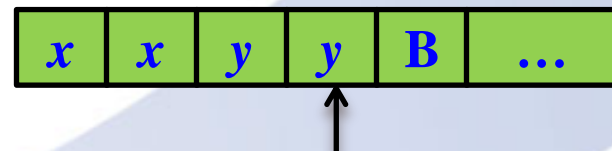
第10步：遇到 x ，右移



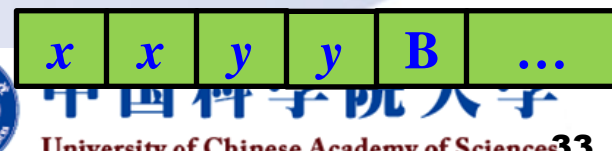
第11步：遇到 y ，右移



第12步：遇到 y ，右移



第13步：遇到B，停机

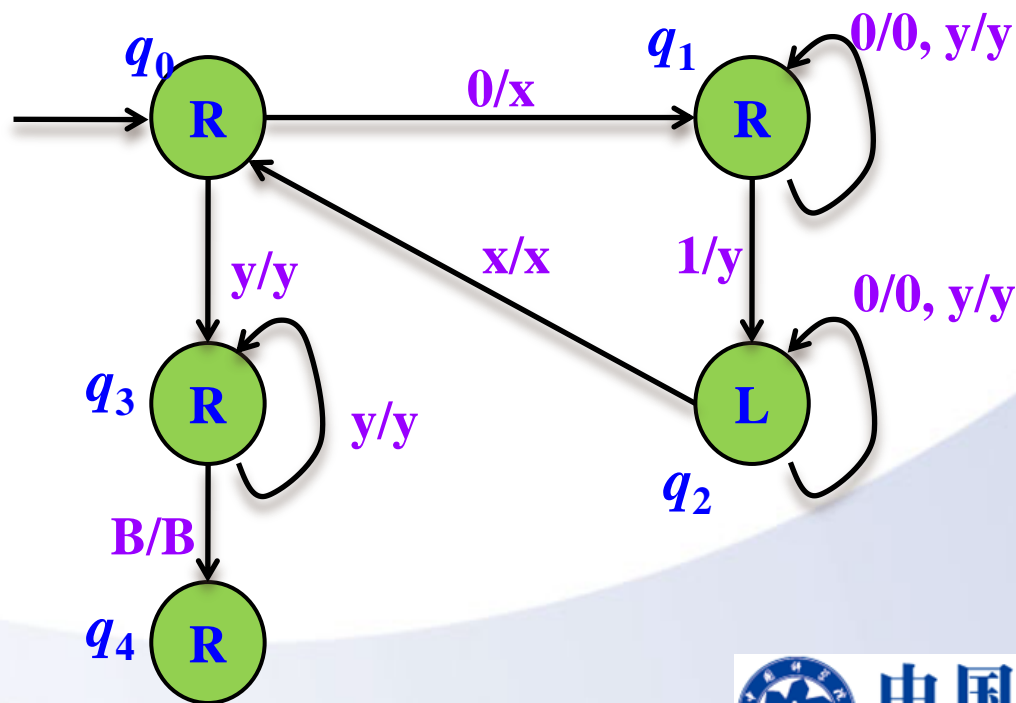


9.1 复杂性分类

■ 2. 图灵机

□ 例3：设计一个图灵机M，它接受的语言 $L=\{0^n1^n \mid n>1\}$ ，设计该图灵机。

➤ 状态转换图



9.1 复杂性分类

■ 2. 图灵机

□ 例3: 设计一个图灵机M, 它接受的语言 $L=\{0^n1^n \mid n>1\}$, 设计该图灵机。

➤ 状态转换图

- q_0 : 初始状态。
- q_1 : 在 q_0 下读到0, 把0改为x, 同时状态改为 q_1 , q_1 下读到0, 不改动, 只右移。
- q_2 : 在 q_1 读到1, 把1改为y, 同时状态改为 q_2 , 向左移。 q_2 下读到0, 不改动, 继续左移, 直到读到x, 状态改为 q_0 。
- q_3 : q_0 下读到y, 状态改为 q_3 。
- q_4 : q_3 下读到B, 状态改为 q_4 , q_4 为终止状态。



9.1 复杂性分类

■ 2. 图灵机

□例3：设计一个图灵机M，它接受的语言 $L=\{0^n1^n \mid n>1\}$ ，设计该图灵机。

➤对应的动作函数 δ 设计如下：

δ	0	1	x	y	B
q_0	(q_1, x, R)	—	—	(q_3, y, R)	—
q_1	$(q_1, 0, R)$	(q_2, y, L)	—	(q_1, y, R)	—
q_2	$(q_2, 0, L)$	—	(q_0, x, R)	(q_2, y, L)	—
q_3	—	—	—	(q_3, y, R)	(q_4, B, R)
q_4	—	—	—	—	—



9.1 复杂性分类

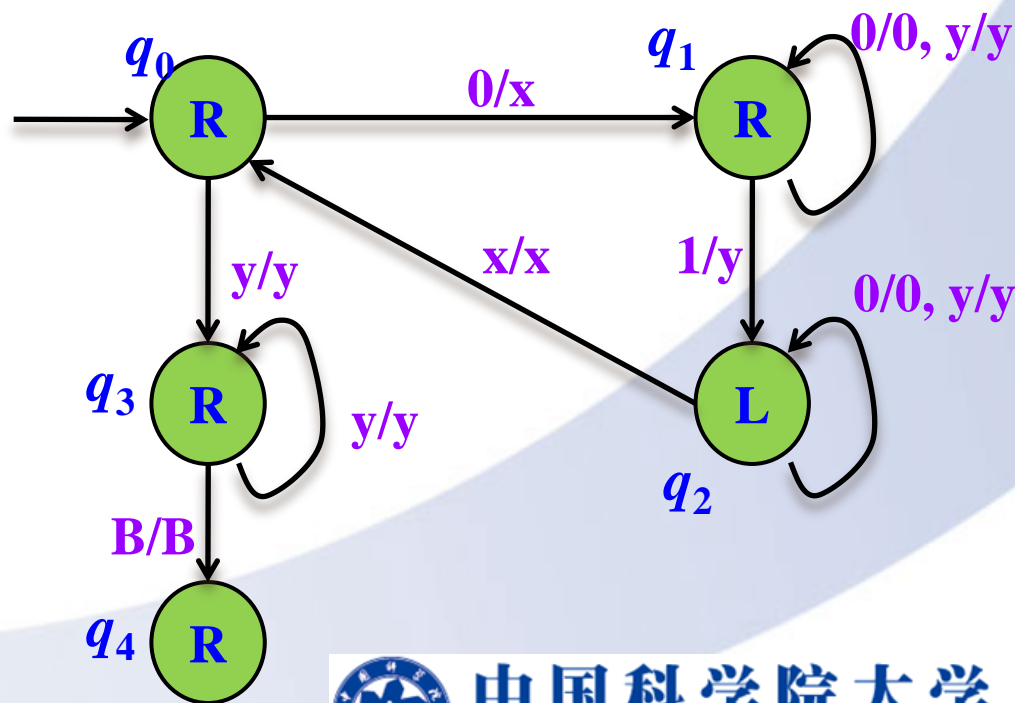
■ 2. 图灵机

□例3：设计一个图灵机M，它接受的语言 $L=\{0^n1^n \mid n>1\}$ ，设计该图灵机。

➤识别输入串 $w_1=0011$ 的瞬像演变过程如下：

$q_00011 \rightarrow xq_1011 \rightarrow x0q_111 \rightarrow$
 $xq_20y1 \rightarrow q_2x0y1 \rightarrow xq_00y1 \rightarrow$
 $xxq_1y1 \rightarrow xxyq_11 \rightarrow xxq_2yy \rightarrow$
 $xq_2xyy \rightarrow xxq_0yy \rightarrow xxyq_3y \rightarrow$
 $xyyyq_4B \rightarrow xxyyBq_4$

进入终止状态 q_4 ，图灵机M停机并接受输入串 w_1 。



9.1 复杂性分类

■ 2. 图灵机

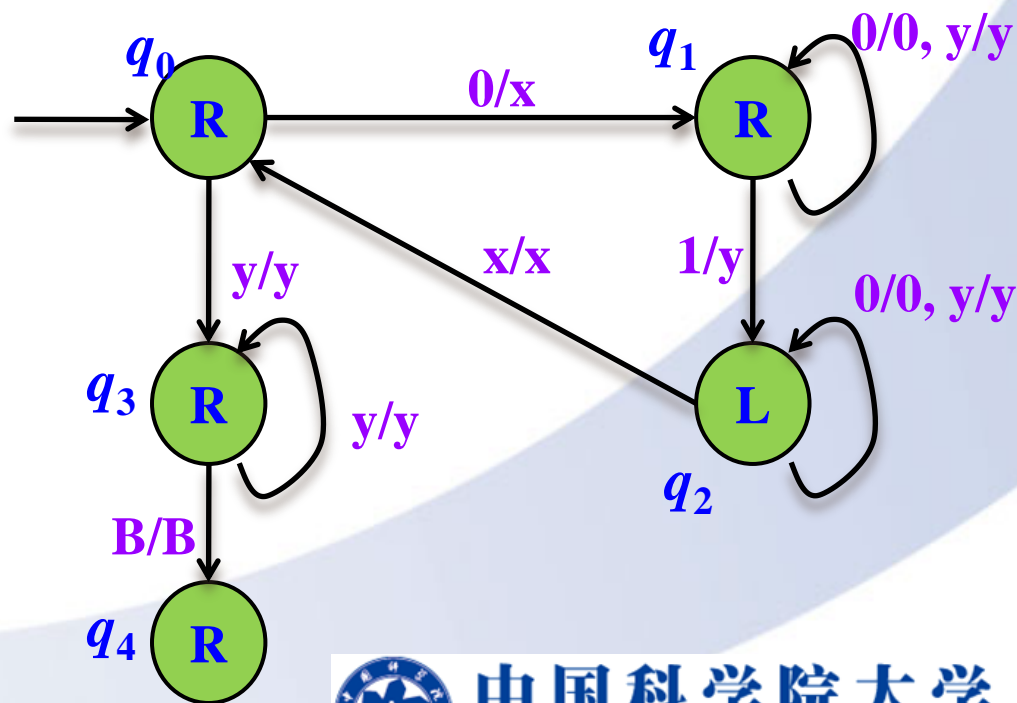
□例3：设计一个图灵机M，它接受的语言 $L=\{0^n1^n \mid n>1\}$ ，设计该图灵机。

➤识别输入串 $w_2=011$ 的瞬像演变过程如下：

$q_0011 \rightarrow xq_111 \rightarrow q_2xy1 \rightarrow$

$xq_0y1 \rightarrow xyq_31$

由于 $\delta(q_3, 1)$ 无定义，而 q_3 不属于终止状态，所以停机但不接受 w_2 。



9.1 复杂性分类

■ 2. 图灵机

□ 从图灵模型我们看到了什么？

- 图灵模型在一定程度上反映了人类**最基本的、最原始的计算能力**，它的基本动作非常简单、机械、确定。

因此，可以用真正的机器来实现图灵机。

- 依据程序，可以对**符合字母表要求的任意符号序列进行计算**。

因此，同一个图灵机可以进行规则相同、对象不同的计算，具有数学概念上的函数 $f(x)$ 的计算能力。



9.1 复杂性分类

■ 2. 图灵机

□ 从图灵模型我们看到了什么？

- 通过组合若干图灵机完成更大更多的计算，如果把一个图灵机对纸带信息变换的结果又输入给另一台图灵机，然后再输入给别的图灵机……，这就是把计算进行了组合。
- 虽然每一个具有特定功能的图灵机很简单，然而当把所有这些图灵机组合在一起，就有可能非常复杂！



9.1 复杂性分类

■ 3. 可计算问题与不可计算问题

□ 图灵结论：

- 凡是可计算的函数，都可以用图灵机计算
- 一个问题是**可计算的**当且仅当它在图灵机上经过**有限步骤**最后得到正确的结果。
- 可计算性 = 图灵可计算性
- 凡是能用算法方法解决的问题，也一定能用图灵机解决；
凡是图灵机解决不了的问题，任何算法也解决不了。



9.1 复杂性分类

■ 3. 可计算问题与不可计算问题

□ 问题分类:

➤ Turing论题把人类面临的所有问题划分成两类:

① 可计算问题

② 不可计算问题

➤ Turing论题中“有限步骤”是一个相当**宽松**的条件，即使需要计算几个世纪的问题，在理论上也都是可计算的。

因此Turing论题界定出的可计算问题几乎包括了人类遇到的所有问题。

水中捞月和大海捞针的区别



9.1 复杂性分类

■ 3. 可计算问题与不可计算问题

□ 不可计算问题的典型例子

① **停机问题**。给定一个计算机程序和一个特定的输入，判断该程序是否可以停机。

✓ 如果停机问题是可计算的，那么编译系统就能够在运行程序之前检查出程序中是否有死循环，

✓ 事实上，当一个程序处于死循环时，系统无法确切地知道它只是一个很慢的程序，还是一个进入死循环的程序。



9.1 复杂性分类

■ 3. 可计算问题与不可计算问题

□ 不可计算问题的典型例子

② 判断一个程序中是否包含计算机病毒。

不存在一个病毒检测程序，能够检测出所有未来的新病毒。

✓ 实际的病毒检测程序做得很好，通常能够确定一个程序中是否包含特定的计算机病毒，至少能够检测现在已经知道的那些病毒，

✓ 但是心怀恶意的人总能开发出病毒检测程序还不能够识别出来的新病毒。



9.1 复杂性分类

■ 4. 易解问题与难解问题

□理论上可计算的问题不一定是实际可计算的。

□Cook论题：

➤一个问题是**实际可计算的**当且仅当它在图灵机上经过**多项式步骤**得到正确的结果。

□易解问题：

➤可以在多项式时间内求解的问题，这类问题在可以接受的时间内实现问题求解。

□难解问题：

➤需要指数时间求解的问题，这类问题的计算时间随着问题规模（输入量的大小）的增长而快速增长，即使中等规模的输入，其计算时间也是以世纪来衡量的。



9.1 复杂性分类

■ 4. 易解问题与难解问题

□ 为什么把多项式时间复杂性作为易解问题和难解问题的分界线呢

① 多项式函数与指数函数的增长率有本质的差别。

表9-1 多项式函数增长和指数函数增长

问题规模n	多项式函数					指数函数	
	$\log_2 n$	n	$n \log_2 n$	n^2	n^3	2^n	$n!$
1	0.00	1	0.00	1	1	2	1
10	3.32	10	33.2	100	1 000	1 024	3 628 800
20	4.32	20	86.4	400	8 000	1 048 376	2.4E18
30	4.91	30	147.2	900	27 000	1.0E9	2.7E32
40	5.32	40	212.9	1 600	64 000	1.0E12	8.2E47
50	5.64	50	282.2	2 500	125 000	1.0E15	3.0E64
100	6.64	100	664.4	10 000	1.0E6	1.3E30	9.3E157



9.1 复杂性分类

■ 4. 易解问题与难解问题

② 计算机性能的提高对多项式时间算法和指数时间算法的影响不同。

表9-2 速度是原来10倍的计算机所能处理问题规模的增长情况

$T(n)$	n	n'	变化	n'/n
$10n$	1,000	10,000	$n'=10n$	10
$20n$	500	5,000	$n'=10n$	10
$5n\log_2 n$	250	1,842	$\sqrt{10} n < n' < 10 n$	7.37
$2n^2$	70	223	$n' = \sqrt{10} n$	3.16
2^n	13	16	$n' = n + \log_2 10 \approx n + 3$	1



9.1 复杂性分类

■ 4. 易解问题与难解问题

- ③ 多项式时间复杂性忽略了系数，但不影响易解问题和难解问题的划分

表9-3 一些特殊的函数的增长率

问题规模 n	多项式函数			指数函数	
	n^8	$10^8 n$	n^{1000}	1.1^n	$2^{0.01n}$
5	390,625	5×10^8	5^{1000}	1.611	1.035
10	10^8	10^9	10^{1000}	2.594	1.072
100	10^{16}	10^{10}	10^{2000}	13,780.612	2
1000	10^{24}	10^{11}	10^{3000}	2.47×10^{41}	1024



9.1 复杂性分类

■ 4. 易解问题与难解问题

□ 类似于 $T_2(n)=10^8n$ 、 $T_3(n)=n^{1000}$ 和 $T_5(n)=2^{0.01n}$ 这样的时间函数是不自然的，在人类遇到的实际问题中，几乎不存在这样的问题以及算法。



作业-课后练习28

■ 课后练习

□ 在图灵机中, 设 **B** 表示空格, q_0 表示图灵机的初始状态, q_F 表示图灵机的终止状态, 如果工作带上的信息为 **B10100010B**, 读写头对准最右边第一个为0的单元, 则按照下指令执行后, 得到的结果是什么? 如果工作带上的信息为 **B10100011B**, 读写头对准右边第一个为1的单元, 则执行指令后得到的结果是什么?

$(q_0 \ 0 \ 1 \ L \ q_1)$ $(q_0 \ 1 \ 0 \ L \ q_2)$ $(q_0 \ B \ B \ H \ q_F)$

$(q_1 \ 0 \ 0 \ L \ q_1)$ $(q_1 \ 1 \ 1 \ L \ q_1)$ $(q_1 \ B \ B \ H \ q_F)$

$(q_2 \ 0 \ 1 \ L \ q_1)$ $(q_2 \ 1 \ 0 \ L \ q_2)$ $(q_2 \ B \ B \ H \ q_F)$



End

