



中国科学院大学
University of Chinese Academy of Sciences

计算机科学导论实验

加法图灵机-1

zxu@ict.ac.cn
zhangjialin@ict.ac.cn

加法图灵机实验安排

周次	日期	课程内容	作业提交内容	截止时间
5	3月28日	讲解实验目标、实验平台、案例	第一版设计	4月11日 10:00
7	4月11日	组内讨论、分享自己的设计	第二版设计	4月18日 10:00
8	4月18日	小组课堂演示	无	

线上实验成绩70%，小组演示30%

线上实验成绩 = $\max(\text{第一版设计}, 0.95 \times \text{第二版设计})$

自由组队

- 共8队，每队6-7人，并选1人作为组长。
- 课后自由组队，助教可做微调。

第1周 提纲

- 图灵机介绍
- 实验平台
- 示例：奇偶判断、回文判断
- 实验内容：三个图灵机加法实验

实验目标

- 设计三个图灵机，分别实现：

- 任意位一进制加法

1	1	1	+	1	1	1	=	1	1	1	1	1	1	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

- 4位二进制加法

0	1	1	1	+	0	1	0	1	=	0	1	1	0	0	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

- 任意位二进制加法

1	1	1	+	1	1	0	1	=	1	0	1	0	0	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

图灵机形式化定义

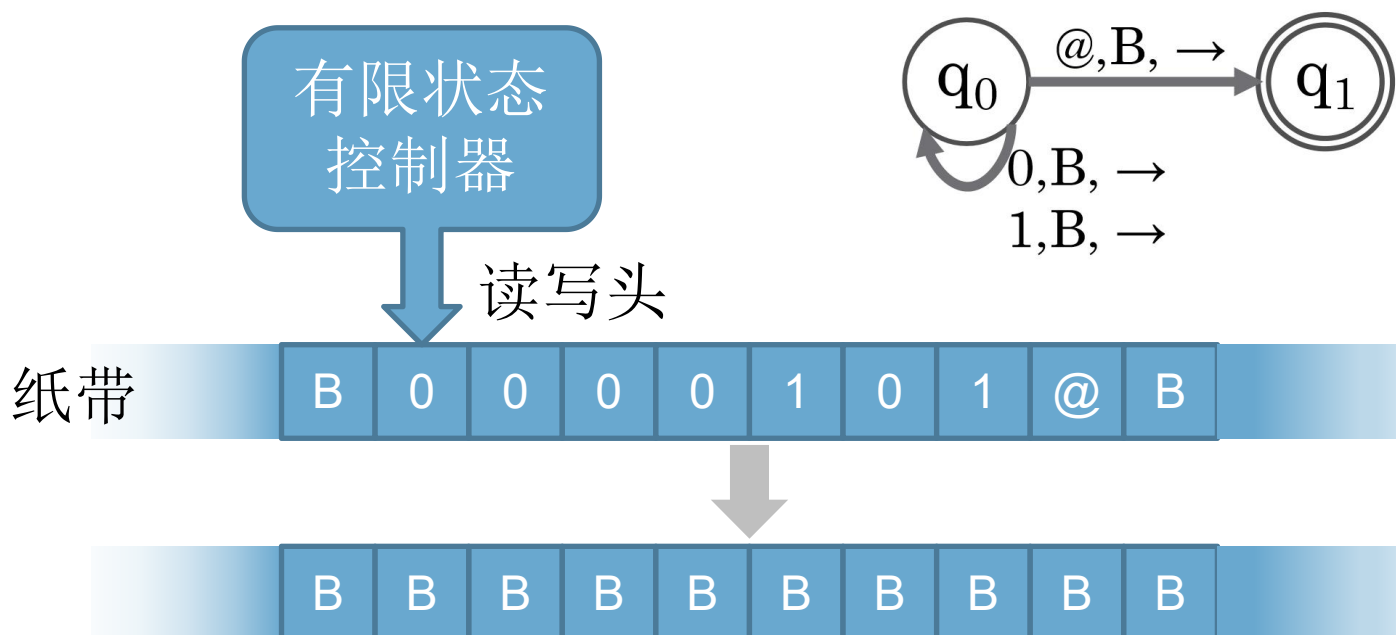
- 图灵机是一个七元组

$$M = \{Q, \Sigma, \Gamma, \delta, q_0, q_{Accept}, q_{Reject}\}$$

- 状态集合 Q
- 输入字母表 Σ
- 带字母表 Γ : 存在特殊字符 $B \in \Gamma$, 表示空白符号, 要求 $B \notin \Sigma$ 且 $\Sigma \subset \Gamma$
- 转移函数 $\delta : (Q - \{q_{Accept}, q_{Reject}\}) \times \Gamma \rightarrow Q \times \Gamma \times \{\rightarrow, \leftarrow\}$
- 起始状态 $q_0 \in Q$
- 接受状态 $q_{Accept} \in Q$
- 拒绝状态 $q_{Reject} \in Q$

图灵机

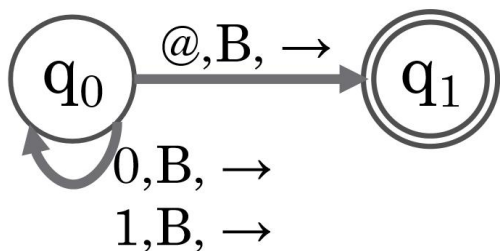
- 图灵机包含一个读写头、有限状态转移图（保存在有限状态控制器中）、无限长纸带
- 清空: 把纸带上的所有符号均用空白符号（**B**）替换
 - 输入字符串要放到纸带上，左右两侧分别紧邻两个空白符号**B**



状态转移图

图灵机

- 图灵机包含一个读写头、有限状态转移图（保存在有限状态控制器中）、无限长纸带
- 清空: 把纸带上的所有符号均用空白符号（**B**）替换
 - 输入字符串要放到纸带上，左右两侧分别紧邻两个空白符号**B**



状态转移图

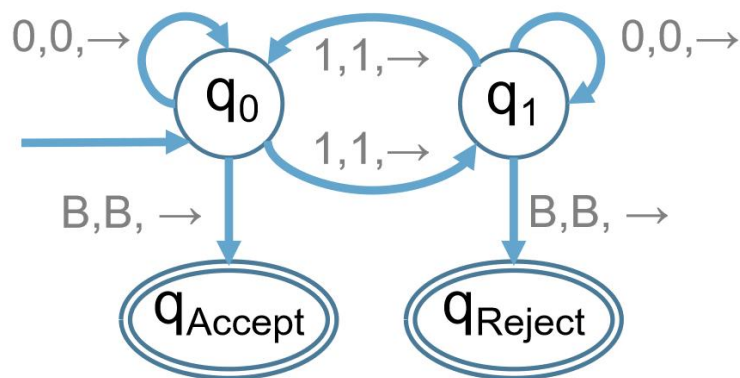
当前状态	读到的符号	写下的符号	读写头移动	下一状态
q_0	0	B	→	q_0
q_0	1	B	→	q_0
q_0	@	B	→	q_1 (Halt)

状态转移表

简单示例

- 判断一个01字符串中“1”的个数是否为偶数
 - 状态集合 $Q = \{q_0, q_1, q_{accept}, q_{reject}\}$
 - 输入字母表 $\Sigma = \{0, 1\}$
 - 带字母表 $\Gamma = \{0, 1, B\}$
 - 转移函数 $\delta: (Q - \{q_a, q_r\}) \times \Gamma \rightarrow Q \times \Gamma \times \{\rightarrow, \leftarrow\}$
 - 见下一页
 - 起始状态 q_0
 - 接受状态 q_{accept} (可简写为 q_a)
 - 拒绝状态 q_{reject} (可简写为 q_r)

简单示例 – 转移函数



$0, 0, \rightarrow$ 表示：
读到0时，写0并右移读写头

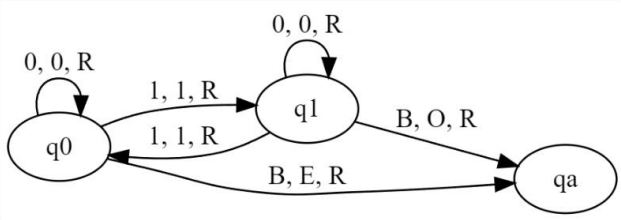
当前状态	读到的符号	写下的符号	读写头移动	下一状态
q_0	0	0	$\rightarrow$	q_0
q_0	1	1	$\rightarrow$	q_1
q_0	B	B	$\rightarrow$	q_{accept}
q_1	0	0	$\rightarrow$	q_1
q_1	1	1	$\rightarrow$	q_0
q_1	B	B	$\rightarrow$	q_{reject}

第1周 提纲

- 图灵机介绍
- 实验平台
- 示例：奇偶判断、回文判断
- 实验内容：三个图灵机加法实验

实验平台

图灵机状态转移图



图灵机模拟动画



当前步数: 0 当前规则: q1, B, O, R, qa 下一规则:

```
1 q0,0,0,R,q0
2 q0,1,1,R,q1
3 q0,B,E,R,qa
4 q1,0,0,R,q1
5 q1,1,1,R,q0
6 q1,B,O,R,qa
```

同学将自己设计的图灵机
状态转移表填入此框

提交规则

下载规则

001011011

设置

继续

单步调试

获取结果

<<

速度: 0.50

>>

获取状态转移图

测试

<https://course.things.ac.cn:10088/>

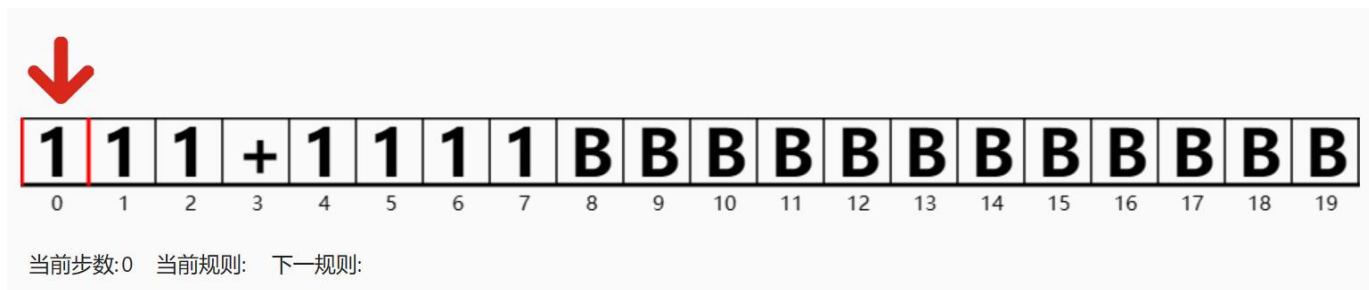
实验平台

- 实验平台说明

- 输入放到纸带上，非输入部分均为字母B
- 两端无限长
- 读写头最开始在输入字符串的最左边

- 停机判断

- 移动步数超过10000步将视作不停机
- 如果在某个输入上模拟超过了10000步，我们认为该图灵机在这个输入上不会停机
- 因此认为没有输出，并认为是错误的



实验平台

● 按钮说明

- **设置**: 初始化图灵机, 导入纸带数据和状态转移表
- **继续/暂停**: 在继续和暂停状态切换图灵机
- **单步调试**: 在暂停阶段, 让图灵机根据当前规则执行下一步
- **获取结果**: 跳过模拟阶段, 直接显示图灵机运行结果
- **<<**: 减速图灵机模拟
- **>>**: 加速图灵机模拟
- **获取状态转移图**: 将当前规则绘制为状态转移图
- **测试**: 查看是否通过一个测试点

提交规则	下载规则							
请输入测试数据 (一个字符)								
设置	继续	单步调试	获取结果	<<	速度: 0.50	>>	获取状态转移图	测试

实验平台：如何使用？

- 只需提交状态转移表

1	q0, B, B, R, q1
2	q1, B, 1, L, qa

- 状态转移表为若干行

- 每行只包含一条转移规则，格式如下：

当前状态,读到的符号,写下的符号,读写头移动,下一状态

(中间是英文逗号)

- 其中：

- 当前状态/下一状态：一个字符串。

- 规定q0为起始状态，qa作为停机状态。

- 读到的符号：一个可见的 **ASCII** 字符，表示读写头读到的符号。

- 写下的符号：一个可见的 **ASCII**字符，表示读写头需要写下的符号。

- 读写头移动：{**L**, **R**} 中的一个，**L** 表示读写头向左移一格，**R** 表示读写头向右移一格。注意要用大写。

- 符号大小写敏感

- 本实验平台不含qr

- 不要有空行

实验平台：如何使用？

- 图灵机以下5部分会由平台自动生成
 - 状态集合： Q 根据状态转移生成，包含状态转移表“当前状态” / “下一状态” 栏中所有出现过的状态。其中规定 q_0 为起始状态， q_a 为停机状态。
 - 带字母表： Γ 规定为所有可见的 **ASCII** 字符的全体。
 - 输入字母表： Σ 规定为 $\Gamma - \{B\}$ ，即集合 Γ 去掉 B 。
 - 起始状态： q_0 。
 - 接受状态： q_a 。

实验平台：示例

- 判断1有奇数个还是偶数个

1	$q_0, \emptyset, \emptyset, R, q_0$
2	$q_0, 1, 1, R, q_1$
3	q_0, B, E, R, q_a
4	$q_1, \emptyset, \emptyset, R, q_1$
5	$q_1, 1, 1, R, q_0$
6	$q_1, B, 0, R, q_a$

第1周 提纲

- 图灵机介绍
- 实验平台
- 示例：奇偶判断、回文判断
- 实验内容：三个图灵机加法实验

示例

● 奇偶判断

- 判断一个01字符串中“1”个数是奇数还是偶数
- 如果是偶数，则在纸带上写下“Even”
- 如果是奇数，则在纸带上写下“Odd”

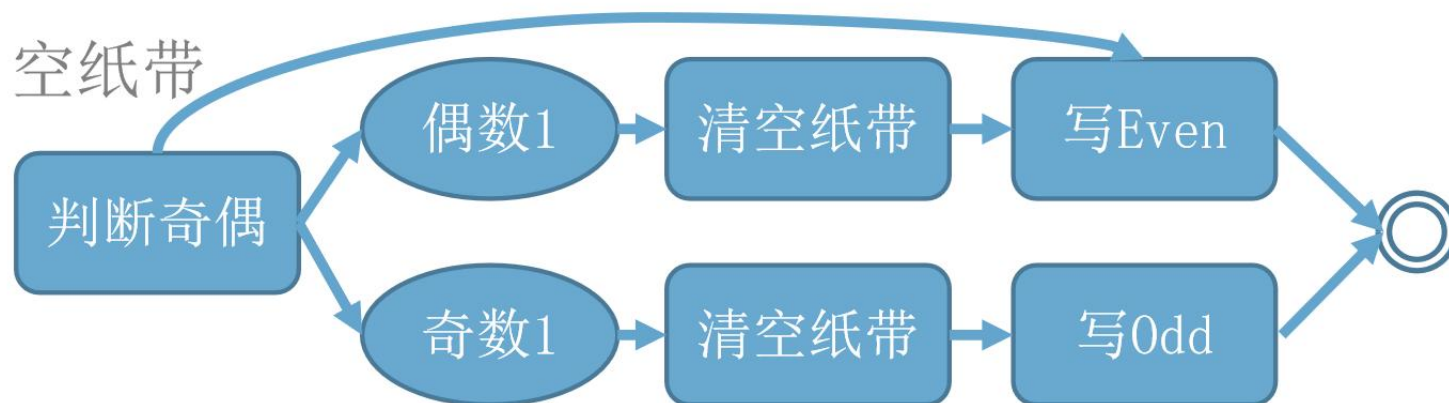
● 回文判断

- 判断一个01字符串是否是回文
- 如果是回文，则在纸带上写下“Y”
- 如果不是回文，则在纸带上写下“N”

示例

- 奇偶判断

- 判断一个01字符串中“1”个数是奇数还是偶数
- 如果是偶数，则在纸带上写下“Even”
- 如果是奇数，则在纸带上写下“Odd”



奇偶判断：思路

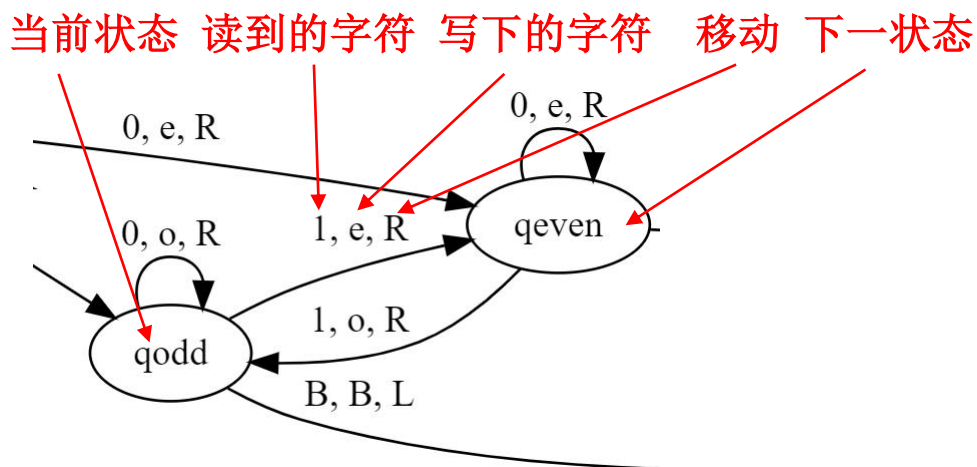
- 状态转移图中的节点

- 状态qodd: 表示在已经读到的字符串中，有奇数个“1”
- 状态qeven: 表示在已经读到的字符串中，有偶数个“1”
- 若读到“1”： $qodd \rightarrow qeven, qeven \rightarrow qodd$
- 若读到“0”： $qodd \rightarrow qodd, qeven \rightarrow qeven$

- 状态转移图中的边

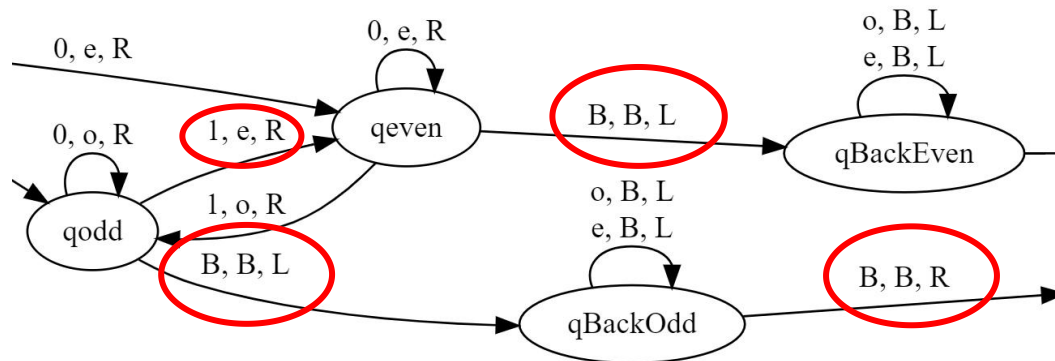
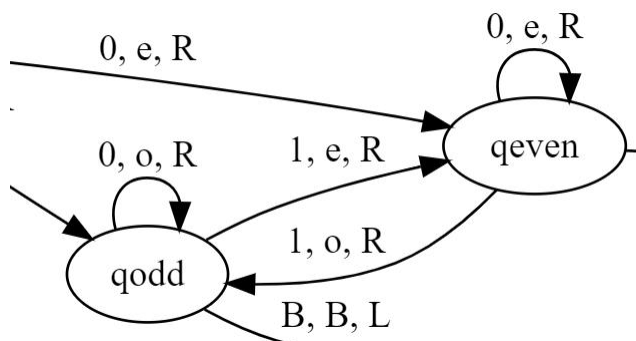
- 以 $qodd \rightarrow qeven$ 为例

< qodd, 1, e, R, qeven >



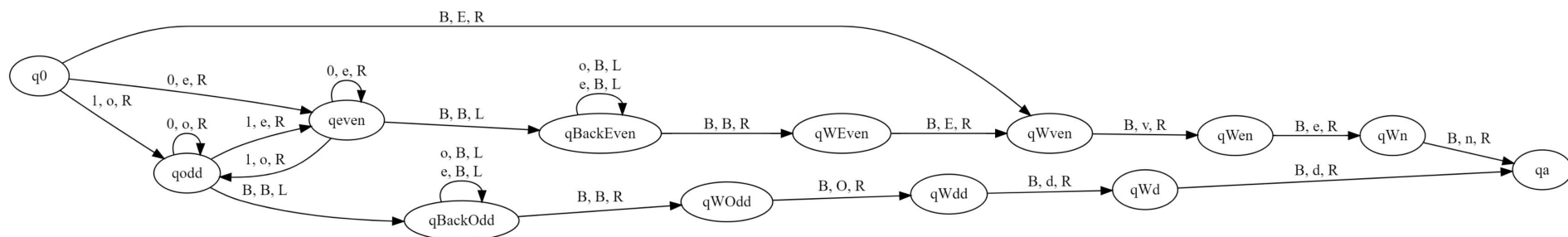
奇偶判断：原理

- 循环：使用4条规则处理任意长输入
 - 而不是使用无穷条规则，例如
 - $q_{\text{odd}}1 \rightarrow q_{\text{even}}1, q_{\text{even}}2 \rightarrow q_{\text{odd}}3, \dots$



- 读写头可以来回移动，能够重复读纸带
 - q_{BackOdd} : 整个字符串有Odd个1，回到起始位置
 - q_{BackEven} : 整个字符串有Even个1，回到起始位置
 - q_{WOdd} : 已在起始位置，准备写字符串“Odd”
 - q_{WEven} : 已在起始位置，准备写字符串“Even”
- 注：起始位置指输入字符串的首字母所在的位置

奇偶判断的状态转移图



qWOdd: 已在起始位置，准备写字符串“Odd”

qWdd: 已写下“O”，准备写字符串“dd”

qWd: 已写下“Od”，准备写字符串“d”

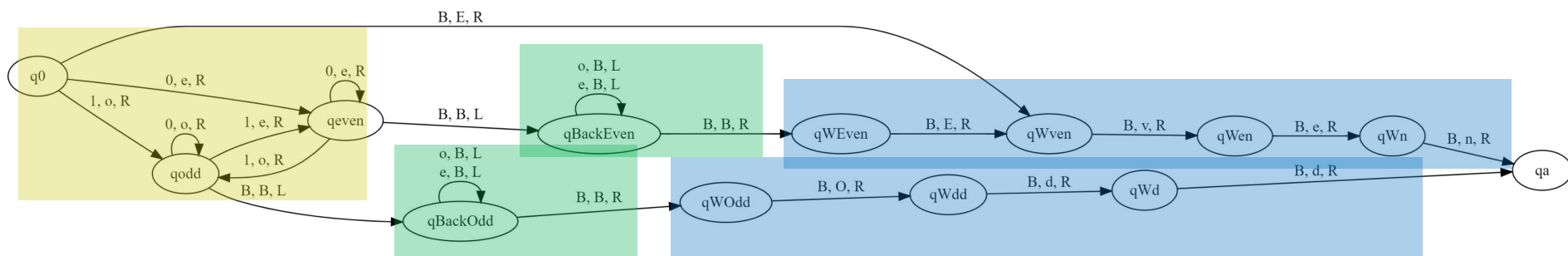
qWEven: 已在起始位置，准备写字符串“Even”

qWven: 已写下“E”，准备写字符串“ven”

qWen: 已写下“Ev”，准备写字符串“en”

qWn: 已写下“Eve”，准备写字符串“n”

奇偶判断的状态转移图



qWOdd: 已在起始位置，准备写字符串“Odd”

qWdd: 已写下“O”，准备写字符串“dd”

qWd: 已写下“Od”，准备写字符串“d”

qWEven: 已在起始位置，准备写字符串“Even”

qWven: 已写下“E”，准备写字符串“ven”

qWen: 已写下“Ev”，准备写字符串“en”

qWn: 已写下“Eve”，准备写字符串“n”

判断奇偶

清空纸带

写入结果

奇偶判断的状态转移规则（状态转移表）

共22行

q0,1,o,R,qodd
q0,0,e,R,qeven
q0,B,e,R,qWven

qWven,B,v,R,qWen
qWen,B,e,R,qWn
qWn,B,n,R,qa

qWOdd,B,O,R,qWdd
qWdd,B,d,R,qWd
qWd,B,d,R,qa

qodd,1,e,R,qeven
qodd,0,o,R,qodd
qodd,B,B,L,qBackOdd

qBackOdd,o,B,L,qBackOdd
qBackOdd,e,B,L,qBackOdd
qBackOdd,B,B,R,qWOdd

qWEven,B,E,R,qWven

qeven,1,o,R,qodd
qeven,0,e,R,qeven
qeven,B,B,L,qBackEven

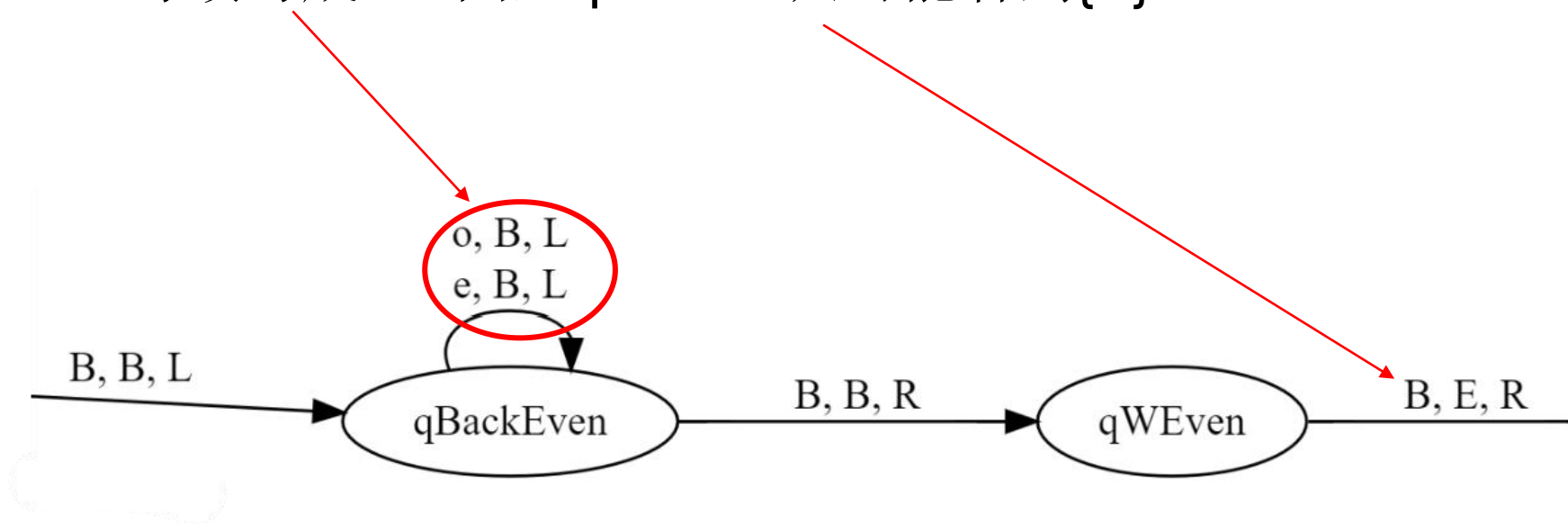
qBackEven,o,B,L,qBackEven
qBackEven,e,B,L,qBackEven
qBackEven,B,B,R,qWEven

奇偶判断的思考题

- 按照图灵机的定义，
 - 状态集 $Q = \{q_0, q_a, q_{odd}, q_{even}, q_{BackEven}, q_{WEven}, q_{Wven}, q_{Wen}, q_{Wn}, q_{BackOdd}, q_{WOdd}, q_{Wdd}, q_{Wd}\}$
 - $|Q| = 13$
 - 带字母表集 $\Gamma = \{0, 1, B, e, o, n, d, v, E, O\}$
 - $|\Gamma| = 10$
 - 转移函数 $\delta : (Q - \{q_a, q_r\}) \times \Gamma \rightarrow Q \times \Gamma \times \{\rightarrow, \leftarrow\}$
 - 定义域决定了状态转移表的行数
 - 状态转移表应有 $|Q - \{q_a, q_r\}| \times |\Gamma| = 12 \times 10 = 120$ 行
- 但我们设计的奇偶判断图灵机的状态转移表只有**22行**！
- 我们设计的奇偶判断图灵机是错的吗？为什么？

奇偶判断的思考题：初步解答

- 对于任意状态，可以忽略读写头不可能读到的字母。
- 例如
 - q_0 : 由于输入字母表是 $\{0,1\}$ ，读写头只可能看到 $\{0,1,B\}$
 - 不可能看到 e,o,n,d,v,E,O
 - q_{WEven} : 在读写头回到起始位置的过程中，会把遇到的字母改写成 B ，因此 q_{WEven} 只可能看到 $\{B\}$



奇偶判断的模拟执行

图灵机模拟动画



1	0	1	1	0	0	0	1	0	1	1	0	0	0	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

当前步数:0 当前规则:qWn, B, n, R, qa 下一规则:

```
1  q0,1,o,R,qodd
2  qWven,B,v,R,qWen
3  qWOdd,B,O,R,qWdd
4  q0,0,e,R,qeven
5  qWen,B,e,R,qWn
6  qWdd,B,d,R,qWd
7  q0,B,E,R,qWven
8  qWn,B,n,R,qa
9  qWd,B,d,R,qa
10 qodd,1,e,R,qeven
11 qBackOdd,o,B,L,qBackOdd
12 qWEven,B,E,R,qWven
13 qodd,0,o,R,qodd
14 qBackOdd,e,B,L,qBackOdd
15 qodd,B,B,L,qBackOdd
16 qBackOdd,B,B,R,qWOdd
17 qeven,1,o,R,qodd
18 qBackEven,o,B,L,qBackEven
19 qeven,0,e,R,qeven
20 qBackEven,e,B,L,qBackEven
21 qeven,B,B,L,qBackEven
22 qBackEven,B,B,R,qWEven
```

提交规则

下载规则

10110001011000

设置

继续

单步调试

获取结果

<<

速度: 1.00

>>

获取状态转移图

测试

示例

● 奇偶判断

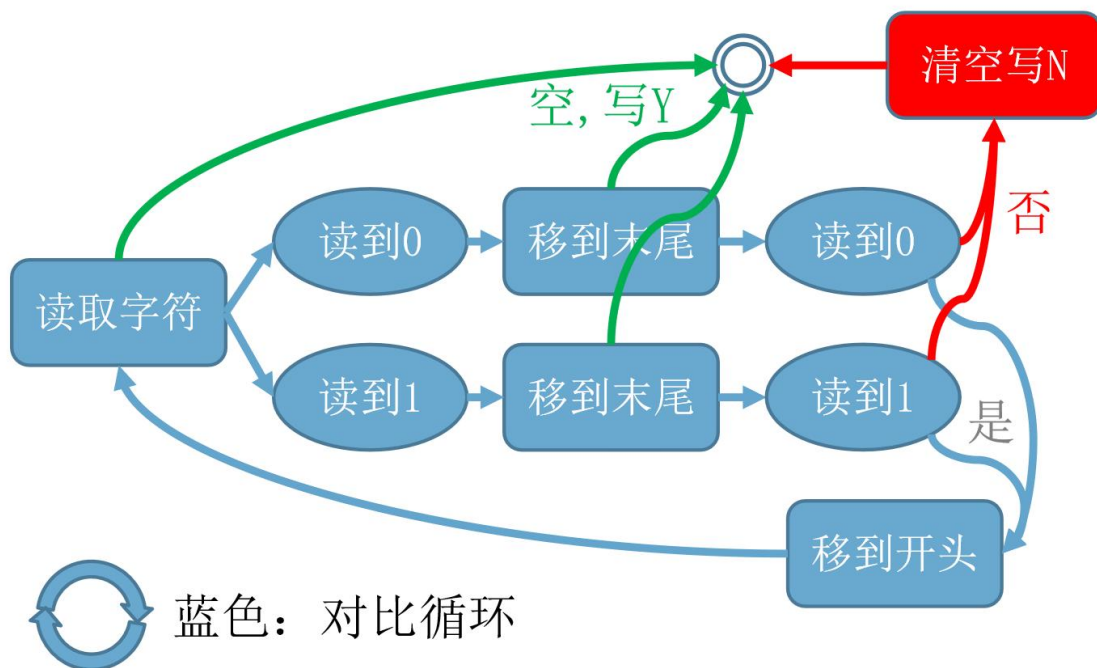
- 判断一个01字符串中“1”个数是奇数还是偶数
- 如果是偶数，则在纸带上写下“Even”
- 如果是奇数，则在纸带上写下“Odd”

● 回文判断

- 判断一个01字符串是否回文
- 如果是回文，则在纸带上写下“Y”
- 如果不是回文，则在纸带上写下“N”

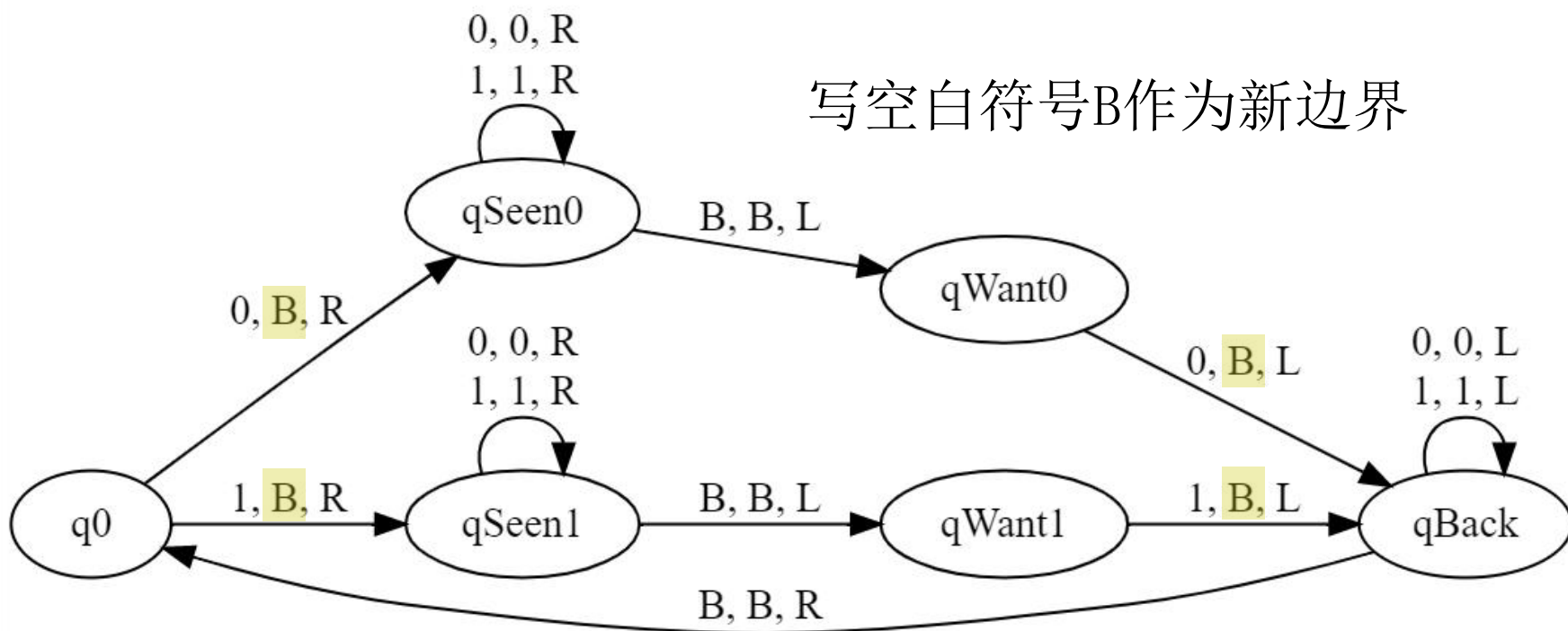
示例

- 回文判断
 - 判断一个01字符串是否回文
 - 如果是回文，则在纸带上写下“Y”
 - 如果不是回文，则在纸带上写下“N”



回文判断：思路

- qSeen0: 字符串剩余部分的最左侧是“0”并右移至边界
- qSeen1: 字符串剩余部分的最左侧是“1”并右移至边界
- qWant0: 期望当前字符为“0”
- qWant1: 期望当前字符为“1”
- qBack: 读写头回到字符串剩余部分的最左侧



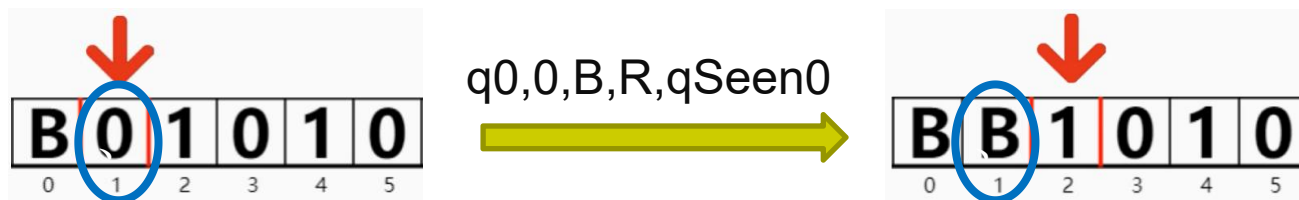
回文判断：原理

● 循环

- 使用有限条规则处理任意长度的输入

● 读写头能来回移动、能做标记

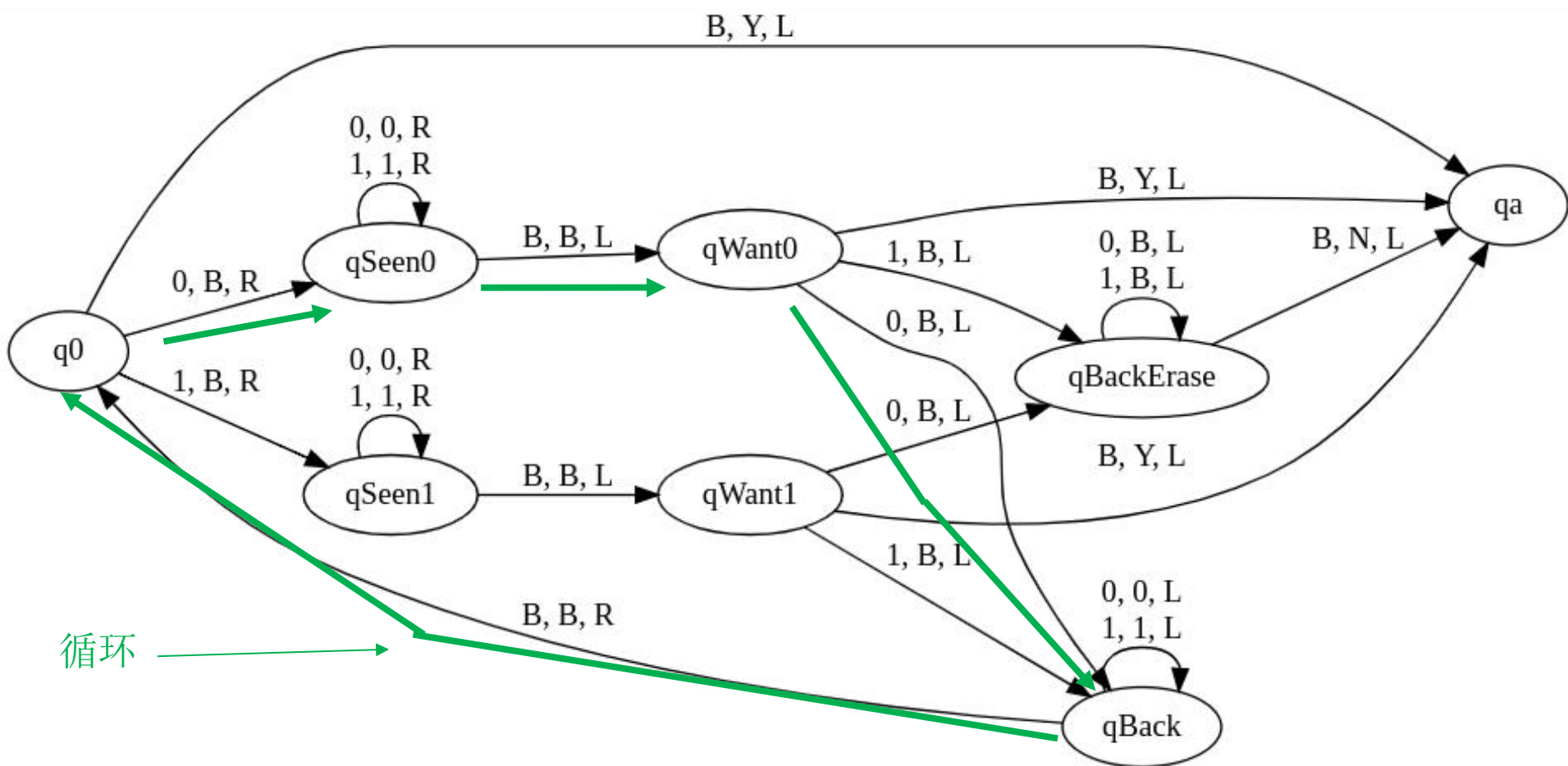
- 可以读纸带任意多次
- 可以向纸带上写下符号（并赋予特定含义）
 - 写下标记符号B，含义是“已看到边界”



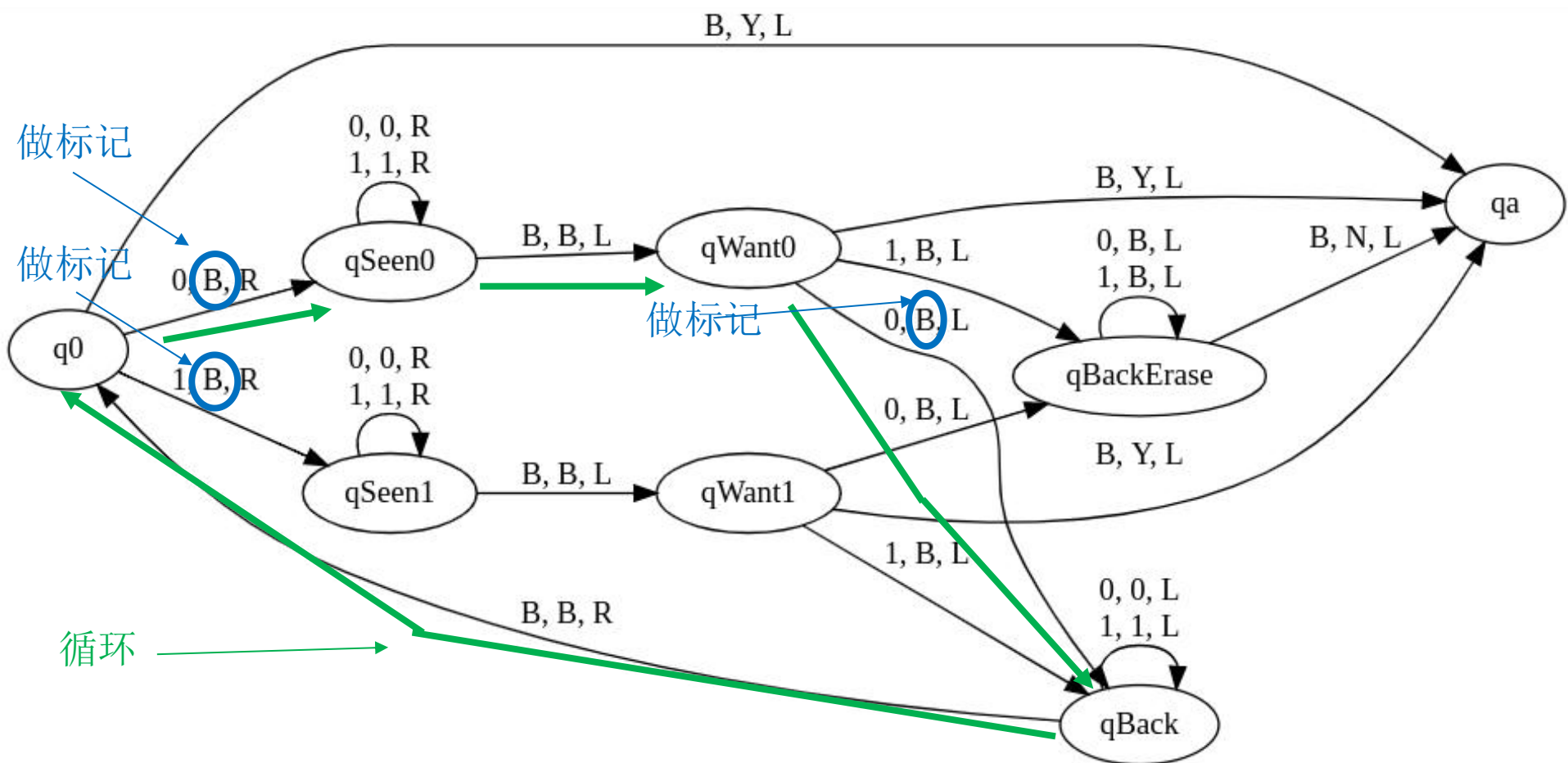
● 隐含状态判断

- 用当前状态和读到的符号作为判断条件，当该条件成立时
 - 写下一个符号，并转移到下一状态

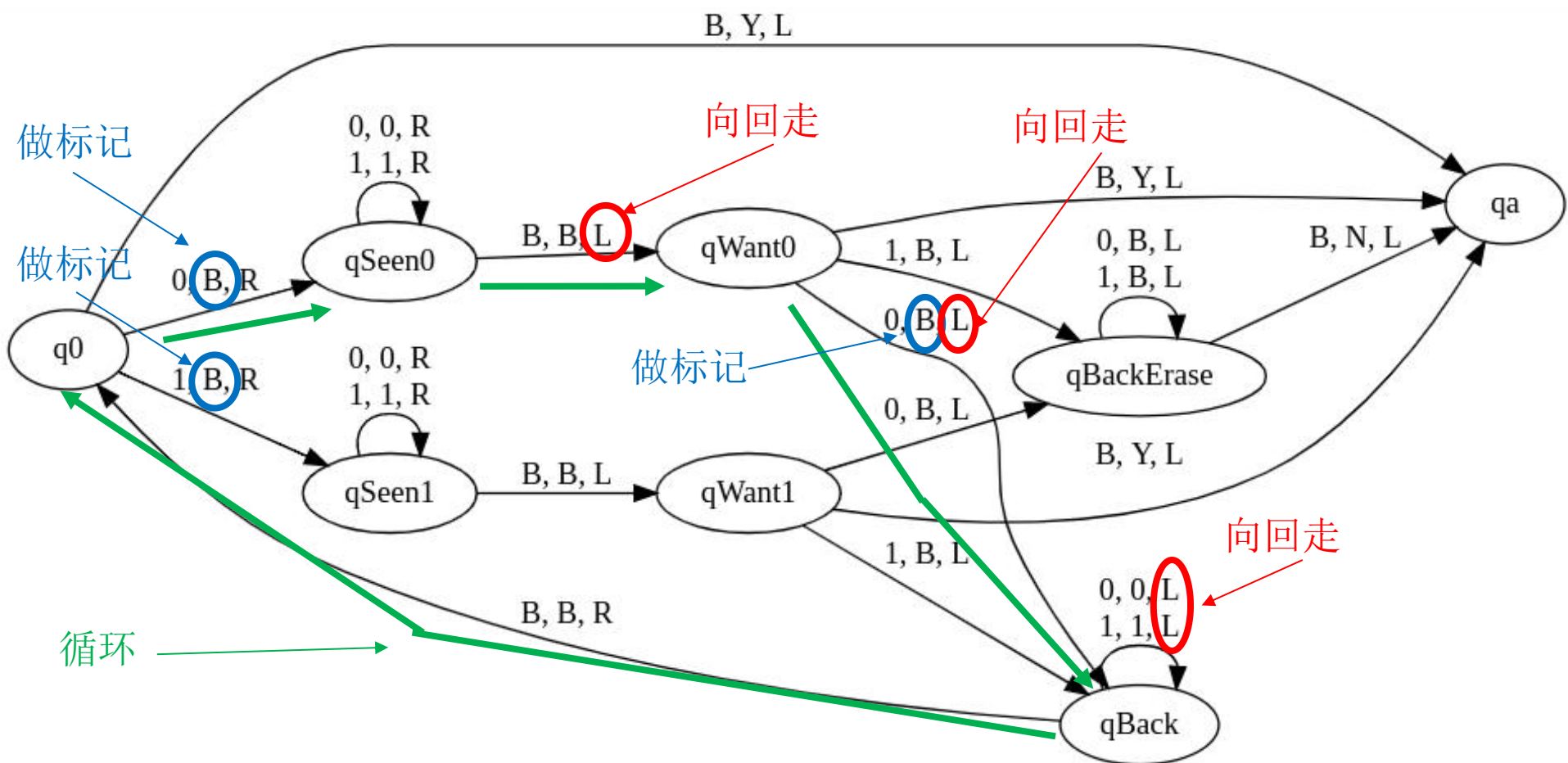
回文判断的状态转移图



回文判断的状态转移图

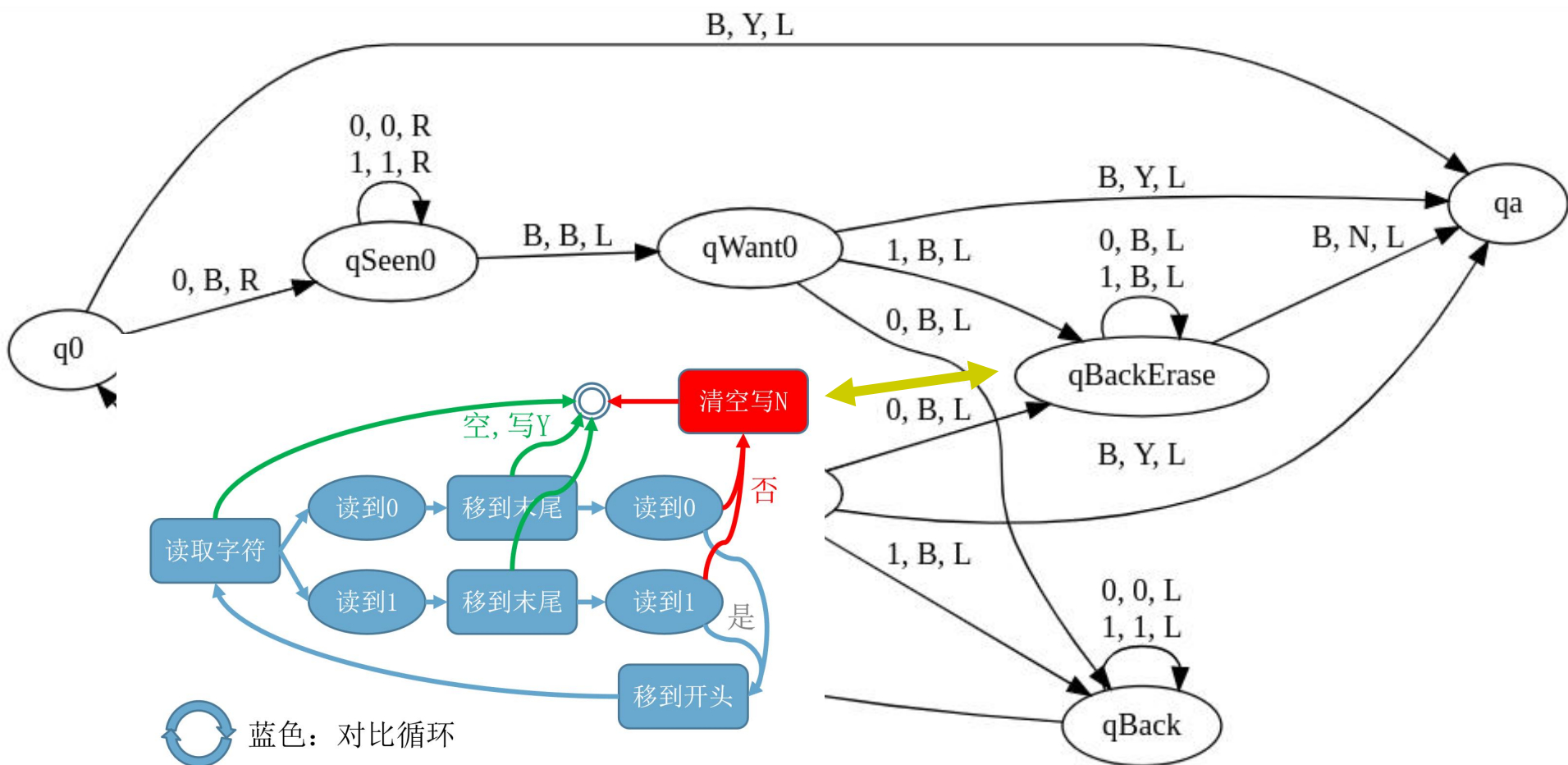


回文判断的状态转移图



回文判断的状态转移图

qBackErase: 读写头写下B, 并回到字符串剩余部分的最左侧



回文判断的状态转移规则

q0,0,B,R,qSeen0	qWant0,0,B,L,qBack	qBackErase,0,B,L,qBackErase
q0,1,B,R,qSeen1	qWant0,1,B,L,qBackErase	qBackErase,1,B,L,qBackErase
q0,B,Y,L,qa	qWant0,B,Y,L,qa	qBackErase,B,N,L,qa
qSeen0,0,0,R,qSeen0	qWant1,0,B,L,qBackErase	
qSeen0,1,1,R,qSeen0	qWant1,1,B,L,qBack	
qSeen0,B,B,L,qWant0	qWant1,B,Y,L,qa	
qSeen1,0,0,R,qSeen1	qBack,0,0,L,qBack	
qSeen1,1,1,R,qSeen1	qBack,1,1,L,qBack	
qSeen1,B,B,L,qWant1	qBack,B,B,R,q0	

回文判断的模拟执行

图灵机模拟动画



0	0	1	0	1	0	0	B	B	B	B	B	B	B	B	B	B	B	B	
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

当前步数:0 当前规则:qWn, B, n, R, qa 下一规则:

```
1  q0,0,B,R,qSeen0
2  qWant0,0,B,L,qBack
3  q0,1,B,R,qSeen1
4  qWant0,1,B,L,qBackErase
5  q0,B,Y,L,qa
6  qWant0,B,Y,L,qa
7  qSeen0,0,0,R,qSeen0
8  qWant1,0,B,L,qBackErase
9  qSeen0,1,1,R,qSeen0
10 qWant1,1,B,L,qBack
11 qSeen0,B,B,L,qWant0
12 qWant1,B,Y,L,qa
13 qSeen1,0,0,R,qSeen1
14 qBack,0,0,L,qBack
15 qSeen1,1,1,R,qSeen1
16 qBack,1,1,L,qBack
17 qSeen1,B,B,L,qWant1
18 qBack,B,B,R,q0
19 qBackErase,0,B,L,qBackErase
20 qBackErase,1,B,L,qBackErase
21 qBackErase,B,N,L,qa
```

提交规则

下载规则

0010100

设置

继续

单步调试

获取结果

<<

速度: 1.00

>>

获取状态转移图

测试

第1周 提纲

- 图灵机介绍
- 实验平台
- 示例：奇偶判断、回文判断
- 实验内容：三个图灵机加法实验

实验内容

- 问题1：两个任意位一进制数加法
- 问题2：两个4位二进制数加法
- 问题3：两个任意位二进制数加法

一进制和二进制加法示例

- 一进制加法
 - $11 + 11 = ?$
 - $111 + 1111 = ?$
- 二进制加法
 - $1010 + 1 = ?$
 - $111 + 1111 = ?$

一进制和二进制加法示例

- 一进制加法

- $11 + 11 = ?$ 1111

- $111 + 1111 = ?$ 1111111

- 二进制加法

- $1010 + 1 = ?$ 1011

- $111 + 1111 = ?$ 10110

问题 1

- 设计一个可以完成任意位一进制数加法的图灵机。
- 格式要求：
 - 输入：(若干1)+(若干1)=
 - 输入的“若干1”表示要做加法的一进制数
 - 可能是零个1
 - 停机时纸带：...###(若干1)B###...
 - 停机时纸带上的“若干1”作为答案
 - 起始位置应为输入的 “=” 位置右侧一格
 - 以 B 作为结尾
 - # 为 Γ 中任意字母

问题 1

- 输入样例

1	1	+	1	=	B	B	B	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13

- 停机时纸带1

1	1	+	1	=	1	1	1	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13

- 停机时纸带2

F	b	c	0	1	1	1	1	B	c	W	W	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13

- 只判定 “=”之后到第一个B之间的输出

问题 2

- 设计一个可以完成4位二进制数加法的图灵机。
- 格式要求：
 - 输入：(4位0或者1)+(4位0或者1)=
 - 输入的“4位0或者1”表示要做加法的二进制数
 - 停机时纸带：...###(5位0或者1)B###...
 - 停机时纸带上的“5位0或者1”作为答案
 - 结果不足5位时注意加**前导0**
 - 起始位置应为输入中“=”位置右侧一格
 - 以 **B** 作为结尾
 - # 为 Γ 中任意字母。

问题 2

- 输入样例1

1	1	0	1	+	0	0	1	1	=	B	B	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

- 输出样例1

1	1	0	1	+	0	0	1	1	=	1	0	0	0	0	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

- 输入样例2

0	0	0	1	+	0	0	1	1	=	B	B	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

- 输出样例2

0	0	0	1	0	0	0	1	1	D	0	0	1	0	0	B	0	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

问题 3

- 设计一个可以完成任意位二进制数加法的图灵机。
- 格式要求：
 - 输入：(若干0或者1)+(若干0或者1)=
 - 输入的“若干0或者1”中不含前导0
 - 每一个操作数的长度大于等于1
 - 停机时纸带：...###(若干0或者1)B###...
 - 停机时纸带上的“若干0或者1”作为答案
 - 不应含有前导0
 - 答案的起始位置应为输入中“=”位置右侧一格
 - 以 B 作为结尾
 - # 为 Γ 中任意字母。

问题 3

- 输入样例

1	1	+	1	0	0	1	=	B	B	B	B	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

- 停机时纸带1

1	1	+	1	0	0	1	=	1	1	0	0	B	B	B	B	B
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

- 停机时纸带2

w	R	Q	T	q	T	L	a	A	1	1	0	0	B	s	R	R	B
-1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

● 纸带双向无限长

评分标准

- 线上实验成绩 (70%)
 - 50%: 任意位一进制数加法
 - 20%: 4位二进制数加法
 - 30%: 任意位二进制数加法
 - $\text{Score} = \text{Max}(\text{第一版分数} * 1, \text{第二版分数} * 0.95)$
- 小组演示(30%)

评分标准：小组演示

- 状态转移表的设计
 - 总结组员的设计方法
- 状态转移表正确性证明
 - 数学证明或实验证明
- 实验中遇到的困难、解决方法
- 实验方案的亮点
- 对图灵机的理解，例如
 - 为什么图灵机能够处理任意长输入
 - 为什么图灵机的计算能力强

注：在课上随机挑选演讲人

6分钟演讲，2分钟补充，2分钟提问。

关键时间节点

- 4月11日10:00前 提交第一次图灵机代码
 - 会给反馈，包括未通过测试点
- 4月18日10:00前 提交第二次图灵机代码
 - 会给反馈，包括未通过测试点、
- 请记录你完成图灵机实验花费的时间



中国科学院大学
University of Chinese Academy of Sciences

计算机科学导论实验

加法图灵机-2

zxu@ict.ac.cn
zhangjialin@ict.ac.cn

课前

- 请按照小组就坐
- 请准备好你的图灵机设计方案

小组讨论

- 组内成员对图灵机设计的相互检查，讨论对图灵机的理解
 - 本次课结束时，每个成员应能够复述其他所有成员的工作
- 建议步骤
 - 每个成员在组内分享自己的设计
 - 记录遇到的bug，难点和关键想法
 - 讨论对图灵机的理解
 - 回答上节课教师提出的问题
 - 准备下次课的课堂小组演示
- 小组组长应担任领导角色
- 请注意第二版设计提交截止时间为4月18日10:00



中国科学院大学
University of Chinese Academy of Sciences

计算机科学导论实验

加法图灵机-3

zxu@ict.ac.cn
zhangjialin@ict.ac.cn

小组课堂演示

- 同组成员相邻就座
- 提前将幻灯片复制到教室电脑
- 随机抽取组内演讲同学
- 演讲要求
 - 演讲者 6分钟上台演讲
 - 同组其他同学 2分钟补充（可选择不补充）
 - 助教或班内其他同学提问
 - 组内任意同学回答问题