第一周

1、丘奇-图灵论题

主要内容：任何在算法上可计算的问题同样可由图灵机计算。

意义：并非定理，而是一个不能严格证明但被普遍接受的命题；

将"可计算性"与"图灵机"等同起来。

2、冯·诺依曼架构 VS 哈佛架构：

两种结构都将计算与访存分开；指令和数据都存放在CPU之外的内存中，由控制器通过总线进行取用。

不同点：前者的指令和数据存储在同一地址空间；

后者将指令存储器和数据存储器分开，从而可以~~并行~~同时调用，避免冲突。

冯·诺依曼架构区分指令和数据的方法：

程序计数器~~中存有~~的地址对应指令

指令中的地址位对应数据（跳转指令除外）

可通过访存操作所在的指令周期进行判断。

3、图灵机

	纸带				状态
(1)	□	0	0	□	q_1
周期1:	□	□	0	□	q_2
2:	□	□	x	□	q_3
3:	□	□	x	□	q_5
4:	□	□	x	□	q_5
5:	□	□	x	□	q_2
6:	□	□	x	□	q_2

7: □ □ x □ q_{accept}

(2) 纸带 状态

	纸带	状态
1:	□ 0 0 0 □	q_1
2:	□ □ 0 0 □	q_2
3:	□ □ x 0 □	q_3
4:	□ □ x 0 □	q_4
5:	□ □ x 0 □	q_{reject}

功能：输入一串"0"，若有偶数个，最终进入 q_{accept}，

否则进入 q_{reject}

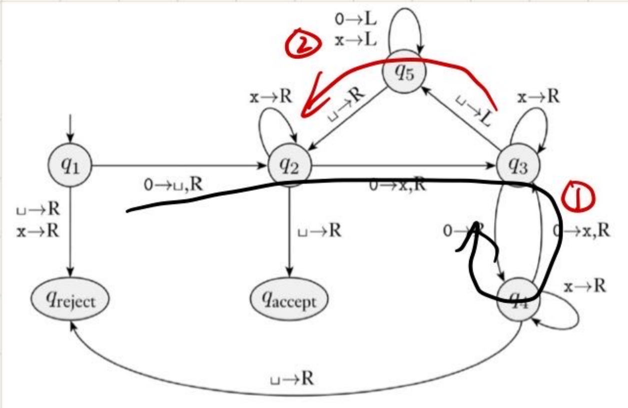
分析：若仅有1/2个0，自不待言

若超过2个0，在读取奇数个0后进入 q_4，此时读到□

将返回 reject

在读取偶数个0后进入 q_3，此时读到□，最终返回 accept.

后经思考，修正如下：



□ 0 0 0 0 0 0 0 0 ···

□ | □ x | 0 x | 0 x | 0 x | ···

end ⇓ accept!　end ⇓ reject!　end ⇓ q_5

（一半的0覆写为x）

□ | □ x x x | 0 x x x | 0 x x x | ···

end ⇓ accept!　end ⇓ reject!　end ⇓ q_5

$$\begin{cases} 2^k \times 3/5/7\cdots & \text{reject!} \\ 2^k \times 2/4/6\cdots & \text{suspect} \\ 2^k \times 1 & \text{accept!} \end{cases}$$

功能：判断0的个数是否为2的整数幂.

周期	纸带	状态
	□ 0 0 0 0 0 0 □	q_1
1	□ □ 0 0 0 0 0 □	q_2
2	□ □ x 0 0 0 0 □	q_3
3	□ □ x 0 0 0 0 □	q_4
4	□ □ x 0 x 0 0 □	q_3
5	□ □ x 0 x 0 0 □	q_4
6	□ □ x 0 x 0 x □	q_3
7	□ □ x 0 x 0 x □	q_5
	遇到0/x向左移	
12	□ □ x 0 x 0 x □	q_5
13	□ □ x 0 x 0 x □	q_2
14	□ □ x 0 x 0 x □	q_2
15	□ □ x x x 0 x □	q_3
16	□ □ x x x 0 x □	q_3
17	□ □ x x x 0 x □	q_4
18	□ □ x x x 0 x □	q_4
19	□ □ x x x 0 x □	q_{reject}