

1.5. 丘奇-图灵论题的主要内容是：任何在算法上可计算的问题同样可由图灵机计算。这个论题表明：所有可以被计算机解决的问题，都可以用类似图灵机这样的通用计算模型来模拟。

丘奇-图灵命题在计算机科学的发展中有着重要的意义。它在真正的计算机出现之前，就揭示出了计算机的本质与其局限性。它说明计算机的能力是有限的。存在一些无法被计算机解决的问题，例如停机问题等。不仅如此，丘奇图灵问题也为早期计算机科学的发展奠定了基础。它提出了一个通用的计算模型：图灵机，为未来计算机的设计与发展产生了重大影响。

1.6. 冯·诺依曼架构奠定了现代计算机的基本结构，哈佛结构正是基于冯·诺依曼架构而来。主要特点与区别：

冯·诺依曼架构：指令和数据存储在同一个内存中，共享一个地址空间；它们采用同一总线传输，因此无法实现指令数据并行读取。

哈佛架构：指令存储器与数据存储器各自有独立的物理地址空间，采用不同的总线结构，因此可以实现并行，速度上较冯·诺依曼结构快。

对于冯·诺伊曼架构，CPU通过程序计数器来跟踪当前执行的指令地址。每执行完一条指令后，计数器会自动指向下一条指令。故处理器能够根据程序计数器指向的地址来判断从内存取得的内容是否是指令。

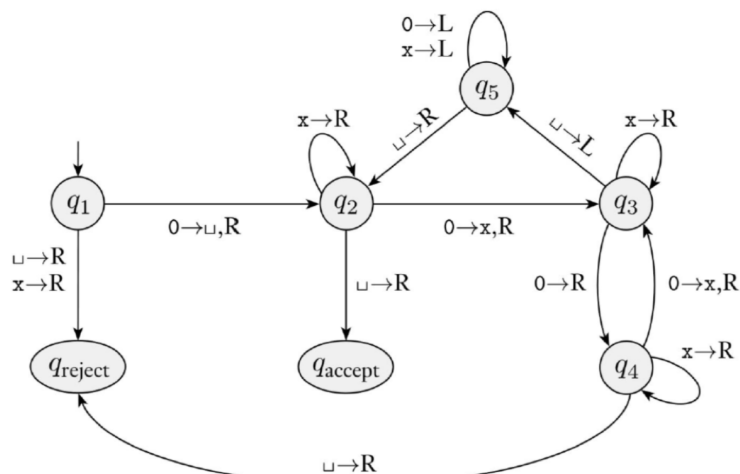
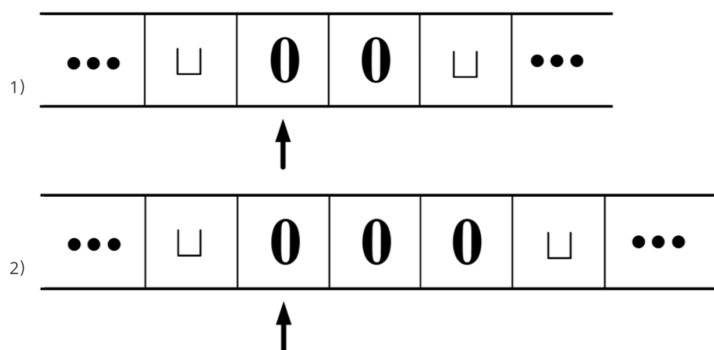
1. 附加题:

已知图灵机的 $K = \{q_1, q_2, q_3, q_4, q_5, q_{accept}, q_{reject}\}$; $\Sigma = 0$; $\Gamma = \{0, x, \sqcup\}$ (其中 \sqcup 表示空白符)

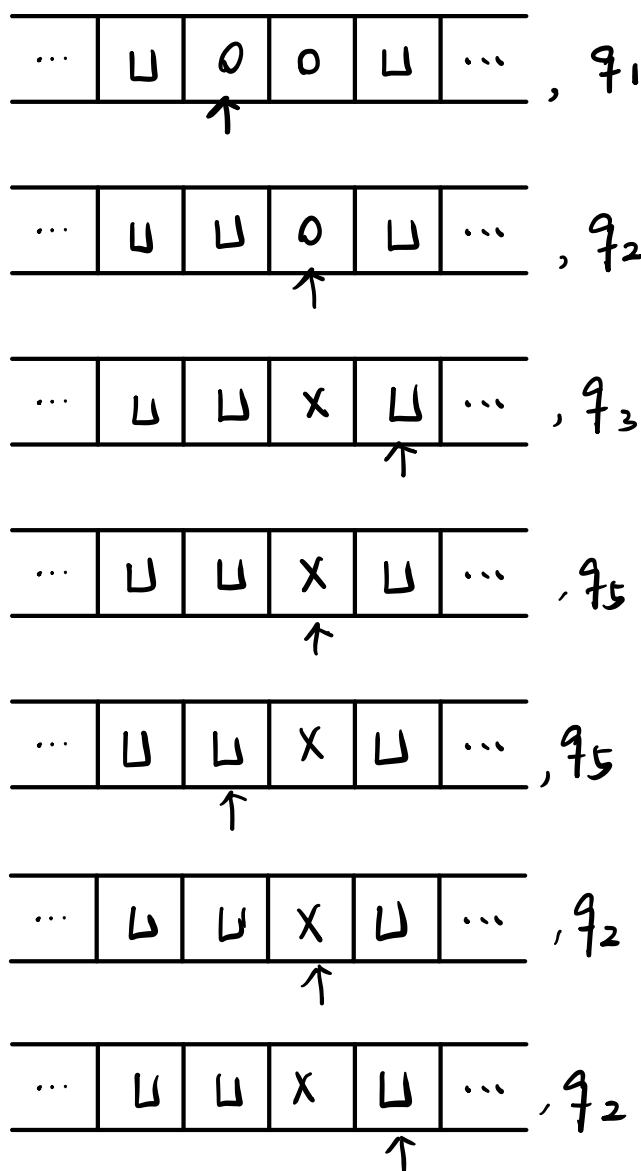
其中 L 代表左移, R 代表右移,

例如: $\sqcup \rightarrow R$ 代表当前纸带为空白符时右移; $\sqcup \rightarrow 0, R$ 代表当前纸带为空白符时, 在纸带上写0再右移

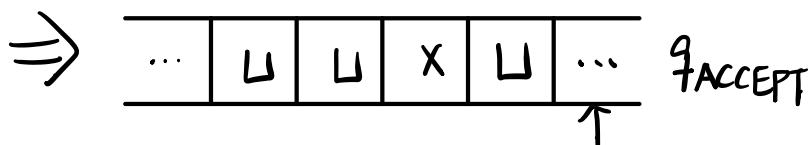
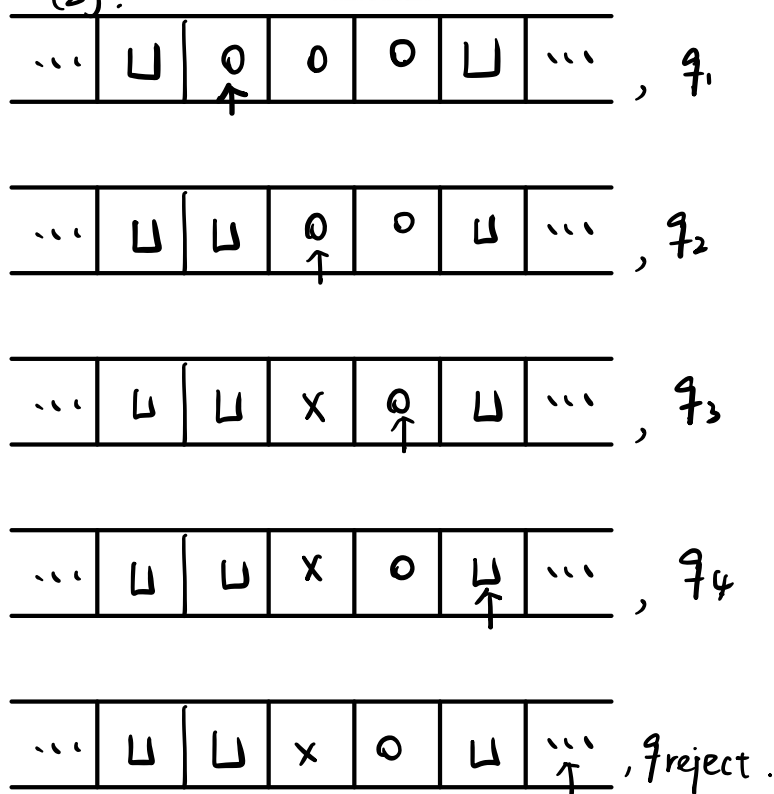
δ 如下图所示



1) 纸带以及状态的变化如下:



(2):



功能：首先，该图灵机检测当前字符是否为“0”，若否，右移且进入 reject 状态；若是，考虑当前位置的“0”为开始的一个最长“0”字符串。

设该字符串长度为 $length$. 考虑它对于 2 的最大幂次 α .

即: $length = 2^a \cdot \beta$, 其中 $\beta \% 2 \neq 0$.

- ① 对于字符串的首个 "0", 图灵机将其写成 "1".
- ② 对于字符串的第 $2^{\alpha+1} \cdot k + 1$ 个位置, 将其保持 "0" 不变, 其中 $k > 0, 2^{\alpha+1} \cdot k \leq \text{length}$
- ③ 字符串的其余位置, 将其写为 "x".
- ④ 最后, 图灵机将指向该字符串

特别的是,若 length 恰为 2 的幂,字符串除第一位将全部被写为 "x",且最后在 q_{accept} 状态结束;否则会停留在 q_{reject} 状态.

利用计算机程序验证: ("—"字符表示"□")

[illegible]