第八章 图灵机

2025年6月19日　　19:08

不可判定的问题

定义：对给定语言$L \subseteq \Sigma^*$和字符串$w \in \Sigma^*$，判断是否$w \in L$的问题，称为语言L上的一个判定性问题(Decision Problem)。如果一个问题, 不存在能解决它的程序, 则称为不可判定问题.

证明：反证法或问题的归约

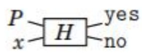
归约：将hello-world问题归约到新问题

hello-world 问题是不可判定的.

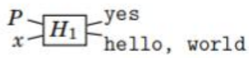
- “给定这个输入的这个程序是否显示 hello, world?”
- 解决这样问题的通用程序是不存在的.

(非形式) 证明: 反证法. 假设这样的程序 H 存在, 它可以在给定程序和输入时, 检查程序的输出是否以 hello, world 开始, 并正确的回答.

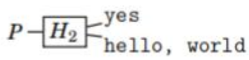
1. H 检查程序 P 在输入 x 时的输出, 并回答 yes 或 no:



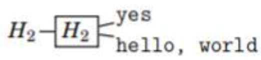
2. 修改 H, 在回答 no 时, 输出 hello, world:



3. 修改 H_1, 将程序 P 作为 P 自己的输入:



4. 那么, 当程序 H_2 以 H_2 为输入时:



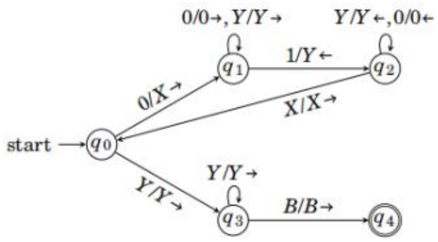
5. H_2 的输出会出现矛盾 (悖论), 所以 H_2 不可能存在, 而从 H 到 H_2 的修改是合理且能行的, 所以 H 不可能存在. □

图灵机TM(Turing Machine)

定义：定义M为七元组$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$，其中Q为有穷状态集，∑为有穷输入符号集，Γ为有穷带符号集且总包含∑，δ为转移函数，且可以对某些自变量无定义，q0为初始状态，B为空白符，F为接受状态集

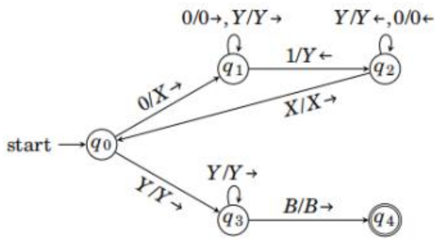
定义：处于状态 q, 带头所在单元格为符号 X, 如果动作的定义为δ(q, X) = (p,Y,L/R), 表示状态转移到 p, 单元格改为 Y, 然后带头向左/右移动一个单元格，这种图灵机称为确定的图灵机

设计识别 $\{0^n1^n \mid n \geq 1\}$ 的图灵机.



$$M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0,1\}, \{0,1,X,Y,B\}, \delta, q_0, B, \{q_4\})$$

设计识别 $\{0^n1^n \mid n \geq 1\}$ 的图灵机.



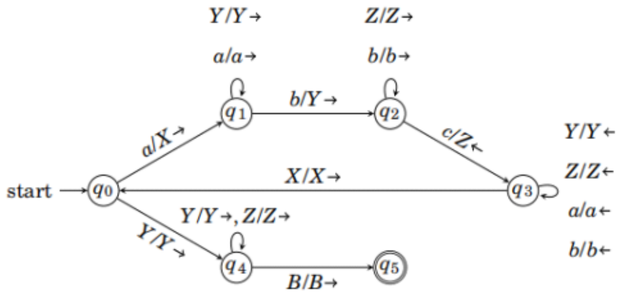
$$M=(\{q_0,q_1,q_2,q_3,q_4\},\{0,1\},\{0,1,X,Y,B\},\delta,q_0,B,\{q_4\})$$

δ	0	1	X	Y	B
$\to q_0$	(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)
$*q_4$	–	–	–	–	–

瞬时描述及其转移：

瞬时描述：$X_1X_z\cdots X_{i-1}qX_iX_{i+1}\cdots X_n$，q为当前状态

设计接受 $L=\{a^nb^nc^n \mid n \geq 1\}$ 的图灵机.



对不接受的字符串，图灵机可能会一直运行

语言与停机：

能够被图灵机接受的语言类，称为递归可枚举的(Recursively Enumerable)

递归语言：对接受和不接受的输入, 都保证停机的图灵机, 所接受的语言称为递归语言

整数函数计算器：图灵机可作为语言的识别器或枚举器, 也可用作整数到整数的函数计算器.M 计算的 f , 不必对所有不同的 i1,i2,...,ik 都有结果.

图灵机的变形

多带图灵机、非确定图灵机、多维图灵机、半无穷带图灵机、多栈机

不考