⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ 作业电子版发送到指定邮箱 ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆

（一）图灵机相关的题目

题目 1：利用字母序对 $\{0,1\}^*$ 中字符串排序，则 $w_1 = \epsilon, w_2 = 0, w_3 = 1, w_4 = 00, w_5 = 01, \ldots$，写出 w_{37} 对应的字符串；如果采用课程所讲授的图灵机编码，请判断该字符串是否代表一个图灵 机，请说明原因。

$\omega_{37}: 00101$

由PPT中的编码，每一个 δ 转移规则都是由4个1分隔的一堆0组成的，而 $\omega_{37}: 00101$ 显然不满足该要求

题目 2：请证明所讲授的onestopTM 与确定单带TM 等价性定理。

下证onestopTM与TM可相互模拟

显然TM可模拟onestopTM（因为onestopTM是TM的特例）

下证onestopTM可模拟TM

设任意TM $M=\left(Q,\Sigma,\Gamma,\delta,q_0,B,F\right)$

其中：$\delta:\left(Q\setminus F\right)\times\Gamma\to Q\times\Gamma\times\left\{L,R\right\}$

我们构造onestopTM $M'=\left(Q',\Sigma,\Gamma,\delta',q_0',B,q_f\right)$

其中：$\delta':\left(Q'\setminus\left\{q_f\right\}\right)\times\Gamma\to Q'\times\Gamma\times\left\{L,R\right\}$

其中：$Q'=\left(Q\setminus F\right)\cup\left\{q_f\right\}$，$q_0'=\begin{cases}q_f & q_0\in F\\ q_0 & q_0\notin F\end{cases}$

δ'：对$\forall p\in Q'\setminus\left\{q_f\right\}\left(=Q\setminus F\right),\forall a\in\Gamma$

$$\delta'\left(p,a\right)=\begin{cases}\left(q_f,a',D\right) & \delta\left(p,a\right)=\left(q,a',D\right)\text{中的}q\in F\\ \left(q,a',D\right) & \delta\left(p,a\right)=\left(q,a',D\right)\text{中的}q\notin F\end{cases}$$

下证$L\left(M'\right)=L\left(M\right)$

下证明，对$\forall\omega\in\Sigma^*$，将其作为M与M'的输入，让它们同时运行，对任意的步数k，M与M'的ID是可以相互确定的。

$(\star)$更形式化地，对任意同一输入$\forall\omega\in\Sigma^*$，对任意步数$\forall k\in\mathbb{N}$，设M有$ID_{k+1}:X_1X_2\cdots X_ipX_{i+1}X_{i+2}\cdots X_n$

，M'有$ID_{k+1}':X_1'X_2'\cdots X_j'qX_{j+1}'X_{j+2}'\cdots X_m'$，要有：

1. $i=j,n=m$
2. $\forall t\left(1\le t\le m\right),X_t=X_t'$
3. $p\in F\Leftrightarrow q=q_f,p\notin F\Leftrightarrow q=p$

下面对运行的步数k进行归纳

① $k=0$，若 M 此时的 ID 为 $ID_1:q_0X_1X_2\cdots X_n$，设 M' 有 $ID_1':q_0'X_1'X_2'\cdots X_n'$

由$k=0$，显然$i=j=-1,n=m,X_1X_2\cdots X_n=X_1'X_2'\cdots X_n'$

由q_0'的构造$q_0'=\begin{cases}q_f & q_0\in F\\ q_0 & q_0\notin F\end{cases}$

有：$q_0\in F\Rightarrow q_0'=q_f,q_0\notin F\Rightarrow q_0'=q_0$

反之，若 M' 此时的 ID 为 $ID_1':q_0'X_1'X_2'\cdots X_n'$，设 M 有 $ID_1:q_0X_1X_2\cdots X_n$

由$k=0$，显然$i=j=-1,n=m,X_1X_2\cdots X_n=X_1'X_2'\cdots X_n'$

由q_0'的构造$q_0'=\begin{cases}q_f & q_0\in F\\ q_0 & q_0\notin F\end{cases}$

有：$q_0'=q_f\Rightarrow q_0\in F,q_0'\ne q_f\Rightarrow q_0'=q_0\Rightarrow q_0\notin F$

结合二者有$q_0\in F\Leftrightarrow q_0'=q_f,q_0\notin F\Leftrightarrow q_0'=q_0$，符合归纳假设

②设$k=m-1$时归纳假设成立，即此时M与M'的ID分别为:

$ID_m:X_1X_2\cdots X_ipX_{i+1}X_{i+2}\cdots X_n$与$ID_m':X_1X_2\cdots X_iqX_{i+1}X_{i+2}\cdots X_n$，

且p与q的关系满足归纳假设。

不妨设$p\notin F$（否则没有第$k+1=m$步的运行），从而根据归纳假设，

$q=p\ne q_f$，故$ID_m=ID_m'$，即:

$ID_m:X_1X_2\cdots X_ipX_{i+1}X_{i+2}\cdots X_n$，$ID_m':X_1X_2\cdots X_ipX_{i+1}X_{i+2}\cdots X_n$

下证 $k=m$ 时归纳假设成立：

(1)$\Rightarrow$，不妨设 $\delta(p,X_{i+1})=(q,X_{i+1}',L)$

若 $q\in F$ ，由 δ' 的构造，知 $\delta'(p,X_{i+1})=(q',X_{i+1}',L)$ ，且 $q'=q_f$

故 $ID_{m+1}:X_1X_2\cdots X_{i-1}qX_iX_{i+1}'X_{i+2}\cdots X_n$ ，

$ID_{m+1}':X_1X_2\cdots X_{i-1}q'X_iX_{i+1}'X_{i+2}\cdots X_n$

这二者满足 $i=j=i-1, n=m=n$ ，

$X_1X_2\cdots X_{i-1}X_iX_{i+1}'X_{i+2}\cdots X_n=X_1X_2\cdots X_{i-1}X_iX_{i+1}'X_{i+2}\cdots X_n$

且有 $q\in F\Rightarrow q'=q_f$

若 $q\notin F$ ，由 δ' 的构造，知 $\delta'(p,X_{i+1})=(q',X_{i+1}',L)$ ，且 $q'=q$

故 $ID_{m+1}:X_1X_2\cdots X_{i-1}qX_iX_{i+1}'X_{i+2}\cdots X_n$ ，

$ID_{m+1}':X_1X_2\cdots X_{i-1}q'X_iX_{i+1}'X_{i+2}\cdots X_n$

这二者满足 $i=j=i-1, n=m=n$ ，

$X_1X_2\cdots X_{i-1}X_iX_{i+1}'X_{i+2}\cdots X_n=X_1X_2\cdots X_{i-1}X_iX_{i+1}'X_{i+2}\cdots X_n$

且有 $q\notin F\Rightarrow q'=q$

(2)$\Leftarrow$，不妨设 $\delta'(p,X_{i+1})=(q',X_{i+1}',L)$

若 $q'=q_f$ ，由 δ' 的构造，知 $\delta(p,X_{i+1})=(q,X_{i+1}',L)$ ，且 $q\in F$

故 $ID_{m+1}:X_1X_2\cdots X_{i-1}qX_iX_{i+1}'X_{i+2}\cdots X_n$ ，

$ID_{m+1}':X_1X_2\cdots X_{i-1}q'X_iX_{i+1}'X_{i+2}\cdots X_n$

这二者满足 $i=j=i-1, n=m=n$ ，

$X_1X_2\cdots X_{i-1}X_iX_{i+1}'X_{i+2}\cdots X_n=X_1X_2\cdots X_{i-1}X_iX_{i+1}'X_{i+2}\cdots X_n$

且有 $q' = q_f \Rightarrow q \in F$

若 $q' \neq q_f$，由 δ' 的构造，知 $\delta\left(p, X_{i+1}\right) = \left(q, X_{i+1}', L\right)$

，且 $q = q'$，从而 $q \notin F$

故 $ID_{m+1} : X_1 X_2 \cdots X_{i-1} q X_i X_{i+1}' X_{i+2} \cdots X_n$，

$ID_{m+1}' : X_1 X_2 \cdots X_{i-1} q' X_i X_{i+1}' X_{i+2} \cdots X_n$

这二者满足 $i = j = i-1, n = m = n$，

$X_1 X_2 \cdots X_{i-1} X_i X_{i+1}' X_{i+2} \cdots X_n = X_1 X_2 \cdots X_{i-1} X_i X_{i+1}' X_{i+2} \cdots X_n$

且有 $q' = q \Rightarrow q \notin F$

综上对 $k = m$，也有 $q \in F \Leftrightarrow q' = q_f, q \notin F \Leftrightarrow q' = q$，归纳假设成立

下证 $L\left(M'\right) = L\left(M\right)$

对 $\forall \omega \in \Sigma^*$，若 $\omega \in L\left(M\right)$，则存在 M 的一个 ID 序列：

$ID_1, ID_2, \ldots, ID_t$ 满足 $ID_1 \vdash_M ID_2 \vdash_M \cdots \vdash_M ID_t$， ID_1 对应了 M 的起始状态 q_0， ID_t 对应了 M 的某个终止状态 $q \in F$

从而根据 $(\star)$，可等价的给出一个 M' 的 ID 序列：

$ID_1', ID_2', \ldots, ID_t'$，且满足 $ID_1' \vdash_{M'} ID_2' \vdash_{M'} \cdots \vdash_{M'} ID_t'$， ID_1' 对应了 M' 的起始状态 q_0'， ID_t' 对应了 M' 的终止状态 q_f

从而 $\omega \in L\left(M'\right)$，从而 $L\left(M\right) \subseteq L\left(M'\right)$

同理有 $L\left(M'\right) \subseteq L\left(M\right)$

故 $L\left(M'\right) = L\left(M\right)$

从而onestopTM与TM等价

题目 3.1：请证明“不允许在带上写入空白符 *B* 的图灵机在计算能力上并没有受到限制”，你需要为新的图灵机给出一个数学定义、将上述直观论述表述为一个定理、证明你的定理。

称“不允许在带上写入空白符B的图灵机”为TM_{NB}，其数学定义可表示为一个七元组：$M_{NB}=\left(Q,\Sigma,\Gamma,\delta,q_0,B,F\right)$，要求$\delta$满足形式：$\delta:\left(Q\setminus F\right)\times\Gamma\to Q\times\left(\Gamma\setminus\left\{B\right\}\right)\times\left\{L,R\right\}$

定理：令M为标准图灵机，$L\left(M\right)$是M所识别的语言，则存在$TM_{NB}:M_{NB}$，满足$L\left(M_{NB}\right)=L\left(M\right)$

证明：

对任意标准图灵机$M=\left(Q,\Sigma,\Gamma,\delta,q_0,B,F\right)$，

根据M可构造$TM_{NB}:M_{NB}=\left(Q,\Sigma,\Gamma\cup\left\{B'\right\},\delta',q_0,B,F\right)$，

其中：

$B'\notin\Gamma$，从而$B'\neq B$，这里的B'作为B的替代字符

对于δ'：对$\forall p\in Q\setminus F$，$\forall a\in\Gamma\cup\left\{B'\right\}$

①对$a\in\Gamma$，若$\delta\left(p,a\right)=\left(q,a',D\right)$

则$\delta'\left(p,a\right)=\begin{cases}\left(q,a',D\right) & 若a'\neq B \quad (A)\\ \left(q,B',D\right) & 若a'=B \quad (B)\end{cases}$

②对$a=B'$，若$\delta\left(p,B\right)=\left(q,a',D\right)$

则$\delta'\left(p,B'\right)=\begin{cases}\left(q,a',D\right) & 若a'\neq B \quad (C)\\ \left(q,B',D\right) & 若a'=B \quad (D)\end{cases}$

先证$L\left(M_{NB}\right)\subseteq L\left(M\right)$:

对$\forall\omega\in L\left(M_{NB}\right)$，存在$M_{NB}$的一个以$\omega$为输入的$ID$序列:

$ID[]=\left[ID_1,ID_2,\cdots,ID_t\right]$，且其满足:

a) ID_1对应了M_{NB}的初始状态q_0，

b) ID_t对应了M_{NB}的某个接受状态$q_f\in F$，

c) 对$\forall i\in\left[1,t-1\right]$，$ID_i\vdash_{M_{NB}} ID_{i+1}$

对$ID[]$进行如下变换：将每个ID中出现的符号B'变为B，不改变ID中的状态，可得到以下的新ID序列:

$ID'[]=\left[ID_1',ID_2',\cdots,ID_t'\right]$

a) 由于ID_1中不存在任何符号B'（由于M_{NB}还没有运行），故有$ID_1'=ID_1$，而变换不改变状态，从而ID_1'对应的状态也为q_0，根据M_{NB}的构造，知ID_1'对应了M的初始状态q_0

b) 由于变换不改变状态，从而ID_t'对应的状态也为q_f，根据M_{NB}的构造，知ID_t'对应了M的一个接受状态q_f

c) 下证，对$\forall i\in\left[1,t-1\right]$，$ID_i'\vdash_M ID_{i+1}'$

不妨设M_{NB}在 $ID_i\vdash_{M_{NB}} ID_{i+1}$的过程中向左移动，根据$M_{NB}$中$\delta'$的构造，有$M$在$ID_i'\vdash_M ID_{i+1}'$的过程中也向左移动。

设$ID_i:\omega_1YpX\omega_2$，$ID_{i+1}:\omega_1qYX'\omega_2$

即有$\delta'\left(p,X\right)=\left(q,X',L\right)$

记变换后的ω_1,ω_2,Y为ω_1',ω_2',Y'

①若$X \neq B', X' \neq B'$，则$ID_i' : \omega_1'Y'pX\omega_2'$， $ID_{i+1}' : \omega_1'qY'X'\omega_2'$

由M_{NB}的δ'构造(A)，有规则：$\delta(p,X)=(q,X',L)$，

从而$ID_i' \vdash_M ID_{i+1}'$

②若$X \neq B', X' = B'$，则$ID_i' : \omega_1'Y'pX\omega_2'$， $ID_{i+1}' : \omega_1'qY'B\omega_2'$

由M_{NB}的δ'构造(B)，有规则：$\delta(p,X)=(q,B,L)$，

从而$ID_i' \vdash_M ID_{i+1}'$

③若$X = B', X' \neq B'$，则$ID_i' : \omega_1'Y'pB\omega_2'$， $ID_{i+1}' : \omega_1'qY'X'\omega_2'$

由M_{NB}的δ'构造(C)，有规则：$\delta(p,B)=(q,X',L)$，

从而$ID_i' \vdash_M ID_{i+1}'$

④若$X = B', X' = B'$，则$ID_i' : \omega_1'Y'pB\omega_2'$， $ID_{i+1}' : \omega_1'qY'B\omega_2'$

由M_{NB}的δ'构造(D)，有规则：$\delta(p,B)=(q,B,L)$，

从而$ID_i' \vdash_M ID_{i+1}'$

综上，对$\forall i \in [1,t-1]$， $ID_i' \vdash_M ID_{i+1}'$

从而$\omega \in L(M)$，从而$L(M_{NB}) \subseteq L(M)$

再证$L(M) \subseteq L(M_{NB})$:

对$\forall \omega \in L(M)$，存在M的一个以ω为输入的ID序列:

$ID[] = [ID_1, ID_2, \cdots, ID_t]$，且其满足:

a) ID_1对应了M的初始状态q_0，

b) ID_t对应了M的某个接受状态$q_f \in F$，

c) 对 $\forall i \in [1, t-1]$， $ID_i \vdash_M ID_{i+1}$

对 $ID[]$ 进行如下变换：将每个 ID 中出现的由 M 写入的符号 B 变为 B'，不改变 ID 中的状态，可得到以下的新 ID 序列：

$ID'[] = [ID_1', ID_2', \cdots, ID_t']$

a) 由于 ID_1 中不存在任何人为写入的符号 B（由于 M 还没有运行），故有 $ID_1' = ID_1$，而变换不改变状态，从而 ID_1' 对应的状态也为 q_0，根据 M_{NB} 的构造，知 ID_1' 对应了 M_{NB} 的初始状态 q_0

b) 由于变换不改变状态，从而 ID_t' 对应的状态也为 q_f，根据 M_{NB} 的构造，知 ID_t' 对应了 M_{NB} 的一个接受状态 q_f

c) 下证，对 $\forall i \in [1, t-1]$， $ID_i' \vdash_{M_{NB}} ID_{i+1}'$

不妨设 M_{NB} 在 $ID_i \vdash_M ID_{i+1}$ 的过程中向左移动，根据 M_{NB} 中 δ' 的构造，有 M 在 $ID_i' \vdash_{M_{NB}} ID_{i+1}'$ 的过程中也向左移动。

设 $ID_i : \omega_1 YpX\omega_2$， $ID_{i+1} : \omega_1 qYX'\omega_2$

即有 $\delta(p, X) = (q, X', L)$

记变换后的 ω_1, ω_2, Y 为 ω_1', ω_2', Y'

①若 $X \neq \text{写入的}B, X' \neq B$，则

$ID_i' : \omega_1'Y'pX\omega_2'$， $ID_{i+1}' : \omega_1'qY'X'\omega_2'$

由 M_{NB} 的 δ' 构造 $(A)(C)$，有规则： $\delta'(p, X) = (q, X', L)$，

从而 $ID_i' \vdash_{M_{NB}} ID_{i+1}'$

②若 $X \neq \text{写入的}B, X' = B$，则

$ID_i':\omega_1'Y'pX\omega_2'$， $ID_{i+1}':\omega_1'qY'B'\omega_2'$

由M_{NB}的δ'构造$(B)(D)$，有规则： $\delta'(p,X)=(q,B',L)$，

从而$ID_i'\vdash_{M_{NB}} ID_{i+1}'$

③若 $X=$写入的$B, X'\neq B$，则

$ID_i':\omega_1'Y'pB'\omega_2'$， $ID_{i+1}':\omega_1'qY'X'\omega_2'$

由M_{NB}的δ'构造(C)，有规则： $\delta'(p,B')=(q,X',L)$，

从而$ID_i'\vdash_{M_{NB}} ID_{i+1}'$

④若 $X=$写入的$B, X'=B$，则

$ID_i':\omega_1'Y'pB'\omega_2'$， $ID_{i+1}':\omega_1'qY'B'\omega_2'$

由M_{NB}的δ'构造(D)，有规则： $\delta'(p,B')=(q,B',L)$，

从而$ID_i'\vdash_{M_{NB}} ID_{i+1}'$

综上，对$\forall i\in[1,t-1]$， $ID_i'\vdash_{M_{NB}} ID_{i+1}'$

从而$\omega\in L(M_{NB})$，从而$L(M)\subseteq L(M_{NB})$

故$L(M_{NB})=L(M)$ （证毕）

题目 4.0：请证明“不允许带头移动到初始位置的左侧的图灵机在计算能力上并没有受到限制”，你需要为新的图灵机给出一个数学定义、将上述直观论述表述为一个定理、证明你的定理。

称“不允许带头移动到初始位置的左侧的图灵机”为$TM_{1/2}$，其数学定义可表示为一个七元组$M_{1/2}=(Q,\Sigma,\Gamma,\delta,q_0,B,F)$，要求带头不允许移动到初始位置的左侧，其他的要求与标准图灵机一致。

定理：令M为标准图灵机，$L(M)$是M所识别的语言，则存在$TM_{1/2}:M_{1/2}$，满足$L(M_{1/2})=L(M)$

证明：

对任意标准图灵机$M=(Q,\Sigma,\Gamma,\delta,q_0,B,F)$，使用多道技术

根据M可构造$TM_{1/2}:M_{1/2}=(Q',\Sigma\times\{B\},\Gamma',\delta',q_0',(B,B),F')$，

其中：

$\Gamma'=\Gamma\times\Gamma\cup\{\#\}$，$\#$符号用于标识$TM_{1/2}$的边界，

$Q'=Q\times\{\uparrow,\downarrow\}$，$\uparrow,\downarrow$用于区分带头处于“上道”还是“下道”，

$q_0'=(q_0,\uparrow)$，

$F'=F\times\{\uparrow,\downarrow\}$

对于δ'：对$\forall p\in Q\setminus F$

①对$\forall a\in\Gamma$，若$\delta(p,a)=(q,a',D)$，$a'\in\Gamma$，$D\in\{L,R\}$

则：对于$\forall b\in\Gamma$：

$$\delta'\big((p,\uparrow),(a,b)\big)=\big((q,\uparrow),(a',b),D\big)$$

$$\delta'\big((p,\downarrow),(b,a)\big)=\big((q,\downarrow),(b,a'),\bar{D}\big)$$

其中，$\bar{D}$表示与D相反的运动方向，

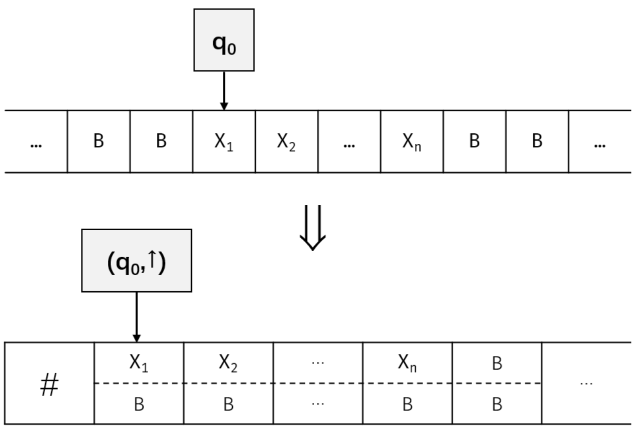
即：$D=L$时$\bar{D}=R$，$D=R$时$\bar{D}=L$

②对于字符#

$$\delta'\big((p,\uparrow),\#\big)=\big((p,\downarrow),\#,R\big)$$

$$\delta'\big((p,\downarrow),\#\big)=\big((p,\uparrow),\#,R\big)$$

对$M_{1/2}$的初始化如下图所示：



更形式化地说，若M初始时的ID为$q_0X_1X_2\cdots X_n$

则$M_{1/2}$初始时的ID为$\#(q_0,\uparrow)(X_1,B)(X_2,B)\cdots(X_n,B)$

仿照上题，可通过两图灵机的ID的互相转换来证明$L(M_{1/2})=L(M)$，这里略去证明。（证毕）

（二）图灵机语言相关的题目

题目 1：证明：如果图灵机 M 判定语言 L，那么 M 识别 L。

证明：由 M 判定语言 L，则 M 始终停机，且 $L(M)=L$，从而 M 识别 L。

题目 2: 证明：若语言 L 是图灵可判定的，则 L 也是图灵可识别的。

证明：由 L 是图灵可判定的，则存在标准图灵机 M， M 判定语言 L，从而 M 识别语言 L，从而 L 是图灵可识别的。

题目 3：证明：如果语言 L 是递归可枚举的，那么存在一个枚举器E，满足 $L(E) = L$ 且 L 中的每个字符串仅被输出一次。

证明:

由L是递归可枚举的，则存在枚举器E_0使得$L\left(E_0\right)=L$

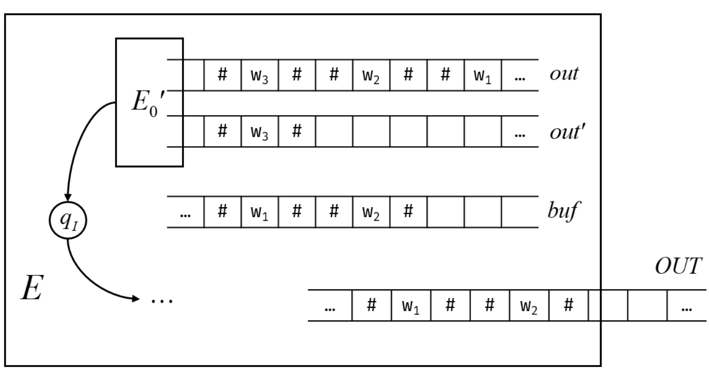
记E_0的原输出带为out，对E_0做出以下可行的拓展:

①为E_0添加一条新的输出带out'，out'默认为空，out'的带头可进行双向的移动，且其移动和读写可由外界控制。当E_0在out进行输出时，同时让E_0在out'进行输出，使得out'的内容与E_0最后一次输出的内容一致。

②当E_0完成一次输出时，不让其返回至常规状态，而是转移到一个处于E_0之外的状态q_I，将控制权转移至E_0外。当外部逻辑结束后，外界再把控制权交回给E_0。

记拓展后的E_0为E_0'

现使用E_0'构建多带图灵机E，如图:



E 包括机器 E_0'，若干工作带，一条缓存带 buf，一条输出带 OUT

当每次 E_0' 完成一次输出，将控制权通过 q_I 交给 E 之后，E 将把 out' 的内容与 buf 中的内容进行比对:

①若 out' 的内容在 buf 中已经存在了，直接清空 out'，将控制权交回给 E_0'；

②若 out' 的内容从未在 buf 中出现，则将 out' 上的内容输出至 OUT 上，并拷贝一份追加至 buf 后。随后清空 out'，将控制权交回给 E_0'。

对于 $\forall\omega\in L(E_0)$，知 E_0 的输出将包含 $\#\omega\#$，而根据 E 的构造，E 只是对 $E_0(E_0')$ 的输出进行了去重的处理，显然 E 的输出也将包含 $\#\omega\#$，从而 $\omega\in L(E)$，从而 $L=L(E_0)\subseteq L(E)$

同样因为 E 只是对 $E_0(E_0')$ 的输出进行了去重的处理，使得 E 的输出都是从 $E_0(E_0')$ 的输出中获得的，从而 $L(E)\subseteq L(E_0)=L$

进而 $L(E_0)=L$

由于 E 的构造，E 始终不输出先前输出过的串，从而该枚举机 E 即符合题意。

题目 4：证明：L 是图灵可判定的 $\Leftrightarrow L$ 和 $\overline{L}$ 都是图灵可识别的。

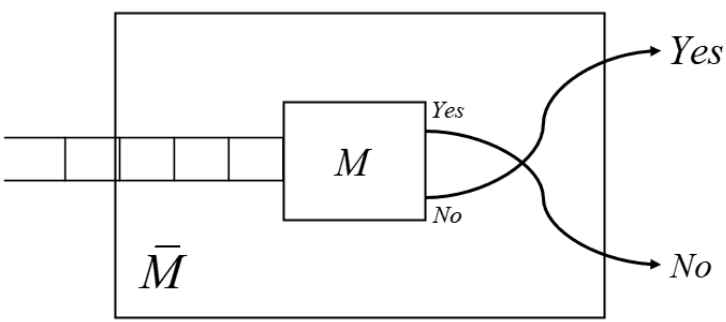
证明:

$(1)\Rightarrow$

由 L 是图灵可判定的，有 L 是图灵可识别的

而由 L 是图灵可判定的，则存在图灵机 M，$L(M)=L$，且 M 始终停机。

从而可构造图灵机 $\bar{M}$，$\bar{M}$ 单纯地把送入 $\bar{M}$ 的输入转发给 M，并将 M 的输出反转后输出。



对 $\forall\omega\in L(\bar{M})$，有 $\bar{M}(\omega)=Yes$，由 $\bar{M}$ 的构造，知 $M(\omega)=No$，从而 $\omega\notin L(M)=L$，即 $\omega\in\overline{L}$，故 $L(\bar{M})\subseteq\overline{L}$；

对 $\forall\omega\in\overline{L}$，有 $\omega\notin L=L(M)$，从而 $M(\omega)=No$，由 $\bar{M}$ 的构造，

有$\bar{M}(\omega)=Yes$，从而$\omega\in L(\bar{M})$，故$\bar{L}\subseteq L(\bar{M})$。

综上，$L(\bar{M})=\bar{L}$，而由于M始终停机，有$\bar{M}$始终停机，从而$\bar{M}$判定$\bar{L}$，故$\bar{L}$也是图灵可判定的，从而$\bar{L}$也是图灵可识别的。

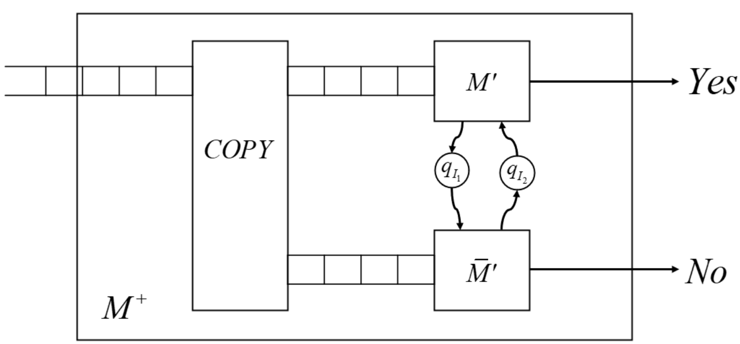
$(2)\Leftarrow$

由于L，$\bar{L}$都是图灵可识别的，从而存在图灵机M，$\bar{M}$，满足$L(M)=L$，$L(\bar{M})=\bar{L}$

对M与$\bar{M}$作出以下改造：

为$M/\bar{M}$分别添加一个外部状态p_{I_1}/p_{I_2}，使得每当$M/\bar{M}$完成执行一次操作后，不在继续执行，而是将状态转移至对应的外部状态p_{I_1}/p_{I_2}。

称改造后的$M/\bar{M}$为$M'/\bar{M}'$，现使用M'与$\bar{M}'$构造图灵机M^+：



M^+在获取输入ω后，将ω送至$COPY$机器中产生两条内容与ω一致的输出，将这两条输出分别送入M'及$\bar{M}'$中，并将控制权交

给M'。

每当M'运行完将状态转移至q_{I_1}后，M^+将控制转交给$\bar{M}'$；每当$\bar{M}'$运行完将状态转移至q_{I_2}后，M^+将控制转交给M'。

这样，M'与$\bar{M}'$将交替地运行。

当M'接收ω时，M^+将停机至接收状态；当$\bar{M}'$接收ω时，M^+将停机至拒绝状态。

下证明M^+判定L：

对$\forall\omega\in L=L(M)$，从而$M\left(M'\right)$将在有限步内接受ω，而$\bar{M}\left(\bar{M}'\right)$将不接受$\omega$，从而$M^+$将停机至接收状态，从而有$\omega\in L\left(M^+\right)$

对$\forall\omega\notin L=L(M)$，有$\omega\in L\left(\bar{M}\right)$，从而$\bar{M}\left(\bar{M}'\right)$将在有限步内接受$\omega$，而$M\left(M'\right)$将不接受$\omega$，从而$M^+$将停机至拒绝状态，从而有$\omega\notin L\left(M^+\right)$

从而$L=L\left(M^+\right)$

而M^+是始终停机的，若否，则$\exists\omega\in\Sigma^*$使得M^+不停机，根据M^+的构造，只可能是$\bar{M}\left(\bar{M}'\right)$与$M\left(M'\right)$均不接收$\omega$，这将分别导致$\omega\notin L$与$\omega\in L$，这是自相矛盾的。

从而M^+判定L，即L是图灵可判定的。

题目 5.1：证明：语言 L 是递归的，当且仅当 L 是图灵可判定的。

证明:

$(1)\Rightarrow$

若L是有穷的，则显然存在图灵机能够判定L。

若L是无穷的，由L是递归的，则存在枚举机E，$L(E)=L$，且E以字典序打印L。

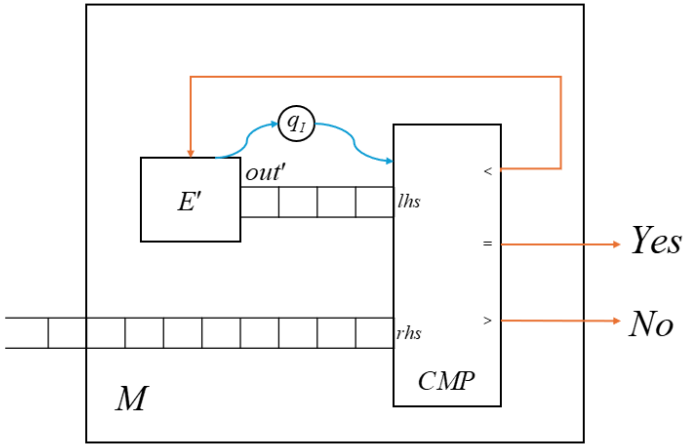
设E原先的输出带为out，为E做出以下可行的拓展：

①为E添加一条新的输出带out'，out'默认为空，out'的带头可进行双向的移动，且其移动和读写可由外界控制。当E在out进行输出时，同时让E在out'进行输出，使得out'的内容与E最后一次输出的内容一致。

②当E完成一次输出时，不让其返回至常规状态，而是转移到一个处于E之外的状态q_I，将控制权转移至E外。当外部逻辑结束后，外界再把控制权交回给E。

记拓展后的E为E'。

下构造图灵机M，如图所示:



M 启动时，将控制权交给 E' 。

每当 E' 生成一个输出 out' ，把状态转移至 q_I 时， M 将把 out' 与 ω 送入 CMP 机器中， CMP 机器将对 out' 与 ω 的字典序进行比较。

根据 CMP 的输出情况， M 将执行对应操作：

①若 $out' < \omega$ ，清空 out' ，把控制权交回 E' 进行下一次输出；

②若 $out' = \omega$ ， M 将停机至接收状态；

③若 $out' > \omega$ ， M 将停机至拒绝状态。

下证 M 判定 L ：

对 $\forall \omega \in L = L(E)$ ，由于 $E(E')$ 以字典序打印 L ，从而 E' 将在有限步内通过 out' 将 ω 输出，这时 CMP 将判定 out' 与 ω 相同，从而 M

将停机至接收状态，即$\omega \in L(M)$

对$\forall \omega \notin L = L(E)$，由于$E(E')$以字典序打印$L$，从而$E'$从不打印$\omega$串。但$E'$将在有限步内打印出一个字典序比$\omega$大的串$\omega'$（若否，即$E'$永不打印字典序比$\omega$大的串，由$E$的定义，这时$L(E)=L$只能是有穷的，与$L$无穷的前提矛盾），这时$CMP$将判定$out'(\omega')$大于$\omega$，从而$M$将停机至拒绝状态，即$\omega \notin L(M)$

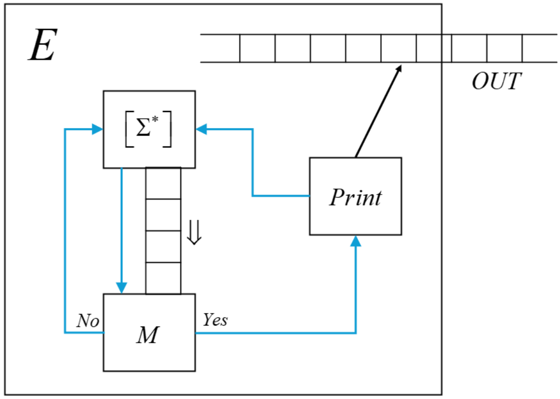
综上，有$L(M)=L$

而M始终停机（若M永不停机，根据M的构造，只可能是CMP始终断言out'小于ω，从而E'始终打印字典序比ω小的串，同先前的论述，这时L有穷，矛盾），从而M判定L，即L是图灵可判定的。

$(2) \Leftarrow$

由L是图灵可判定的，则存在图灵机M，$L(M)=L$且M始终停机。

使用M来构建枚举机E，如图所示:



E 中有机器 $[\Sigma^*]$，该机器将按照字典序在其输出带上打印出属于 Σ^* 的每一个串（即：$\epsilon,0,1,00,01,10,11,000,\cdots,111,0000,\cdots$）

当 $[\Sigma^*]$ 每输出完一个串后，将控制权交给 M，M 将对 $[\Sigma^*]$ 的本次输出进行判定：

①若 M 判定结果为 Yes，则 E 将调用机器 $Print$ 将 $[\Sigma^*]$ 的本次输出追加打印至对外输出带 OUT 上，打印后把控制权交回 $[\Sigma^*]$，让 $[\Sigma^*]$ 进行下一次输出；

②若 M 判定结果为 No，则 E 直接把控制权交回 $[\Sigma^*]$，让 $[\Sigma^*]$ 进行下一次输出。

由于 E 会输出且只会输出 Σ^* 中使得 M 判定为 Yes 的串，易见 $L(E)=L(M)=L$。且由于 $[\Sigma^*]$ 是按照字典序输出的，从而使得 E 也是按照字典序打印输出。故 L 是递归的。

题目 6.3：证明：图灵可识别和可判定语言在**闭包操作**下封闭。

证明:

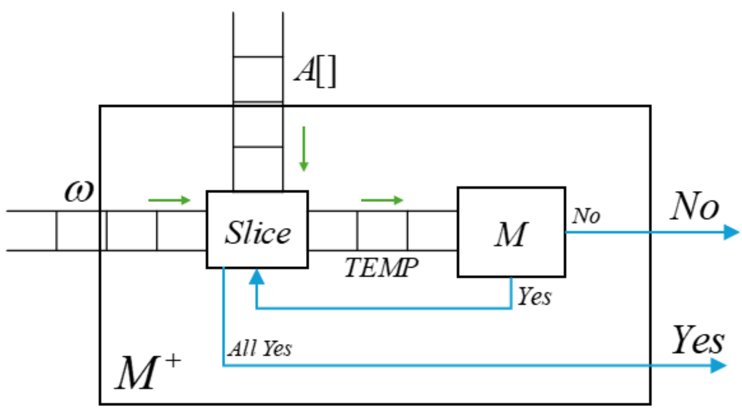
现定义字符串 ω 的*划分数组* $A[]$： $A[]=[a_1,a_2,\cdots,a_s]$，要求 $s\geq 1$，$a_i\geq 1(1\leq i\leq s)$ 且 $\sum_{i=1}^{s}a_i=|\omega|$。

先证明图灵可判定语言在闭包操作下封闭:

设语言 L 是图灵可判定的，则存在图灵机 M，$L(M)=L$，且 M 始终停机。

要证 L^* 也是图灵可判定的，只要证存在图灵机 M^*，$L(M^*)=L^*$，且 M^* 始终停机。

欲构造图灵机 M^*，我们先构造这样的图灵机 M^+，如图所示:



M^+ 将字符串 ω 与划分数组 $A[]$ 作为输入，执行以下操作:

①拷贝一份 ω，记为 ω'

②从左到右逐项读取 $A[]$ 中的元素 a：

将 ω' 前 a 个字符拷贝至 $TEMP$ 带，并从 ω' 中删去这 a 个字符

（即执行切片操作，在图中示意为 $Slice$ 机器）

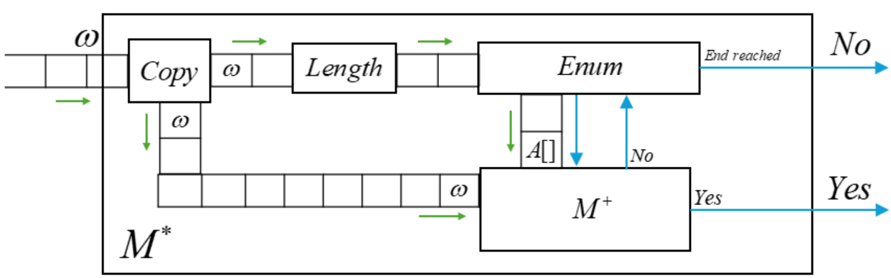
将 $TEMP$ 送入机器 M 中:

当 M 返回 Yes 时，对 $A[]$ 进行下一项的读取遍历

否则当 M 返回 No 时，M^+ 输出 No

③当 $A[]$ 遍历完毕而 M 尚未输出 No 时（即 M 所有返回值都是 Yes），M^+ 输出 Yes

现构造图灵机 M^*，如图所示:



M^* 将外界输入 ω 送至 $Copy$ 中，产生两条内容与 ω 一致的输出，分别送入 $Length$ 与 M^+ 中。

$Length$ 将计算 ω 的长度，送入 $Enum$ 中。

$Enum$ 将根据输入的数值，对长度进行所有可能情况的划分（不重不漏地）并进行输出（例：对于输入 3， $Enum$ 将分别输出 $[3]$，$[1,2]$，$[2,1]$，$[1,1,1]$）。当 $Enum$ 每完成一次输出 $A[]$，就将 $A[]$ 送入 M^+ 并将控制权交给 M^+。

当 M^+ 输出 Yes 时， M^* 立即输出 Yes；当 M^+ 输出 No 时，将控制权交给 $Enum$ 进行下一次划分的生成。

若 $Enum$ 穷尽了所有可能的划分且被再次调用， M^* 输出 No 。

对 $\forall \omega \in L\left(M^*\right)$，可知对 ω 存在一个由 $Enum$ 生成的划分数组 $A[]=[a_1,a_2,...,a_s]$，使得 M^+ 将 ω 与 $A[]$ 作为输入时，其输出为 Yes，从而使 M^* 返回 Yes 。

由 M^+ 的构造知，存在字符串列 $\omega_1,\omega_2,\cdots,\omega_s$， $\left|\omega_i\right|=a_i$，$\omega=\omega_1\omega_2\cdots\omega_s$，这些 ω_i 对 M 的输出都为 Yes，即 $\forall i\in[1,s],\omega_i\in L\left(M\right)=L$，从而 $\omega=\omega_1\omega_2\cdots\omega_s\in L^s\subseteq L^*$，即 $\omega\in L^*$

对 $\forall\omega\in L^*$，知 ω 有划分 $\omega=\omega_1\omega_2\cdots\omega_s\in L^s\subseteq L^*$，其中 $s\geq 1$，$\left|\omega_i\right|\geq 1$， $\omega_i\in L$

从而将 $A[]=\left[\left|\omega_1\right|,\left|\omega_2\right|,\cdots,\left|\omega_s\right|\right]$ 与 ω 作为 M^+ 的输入时， M^+ 将输出 Yes 。而根据 $Enum$ 的构造， $Enum$ 总将在有限步内将上述的 $A[]$ 作为其输出，这时，由上所述， M^+ 将输出 Yes，从而 M^* 输出 Yes，

即有$\omega \in L\left(M^*\right)$。

从而$L\left(M^*\right)=L^*$，而对于给定的串ω，M^*中用到的机器都将在有限步内停机，从而M^*也将始终停机，故M^*判定L^*。

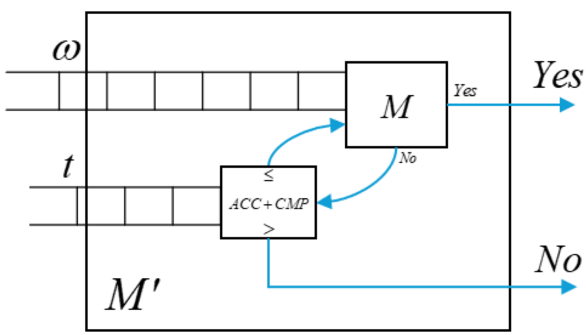
再证明图灵可识别语言在闭包操作下封闭：

设语言L是图灵可识别的，则存在图灵机M，$L(M)=L$

要证L^*也是图灵可识别的，即证存在图灵机M^*，$L\left(M^*\right)=L^*$

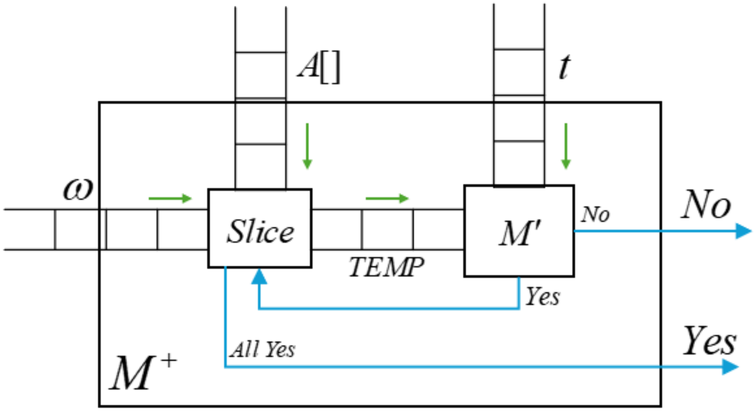
欲构造图灵机M^*，需构造下列图灵机M'与M^+：

先构造M'，如图所示：



M'将字符串ω与运行步数t作为输入，一步一步地执行M（以ω为输入）。若M在t步之内返回Yes，则M'返回Yes，否则M'返回No。

再构造 M^+，如图所示:



M^+ 将字符串 ω、划分数组 $A[]$ 与执行步数 t 作为输入，执行以下操作:

①拷贝一份 ω，记为 ω'

②从左到右逐项读取 $A[]$ 中的元素 a：

将 ω' 前 a 个字符拷贝至 $TEMP$ 带，并从 ω' 中删去这 a 个字符

（即执行切片操作，在图中示意为 $Slice$ 机器）

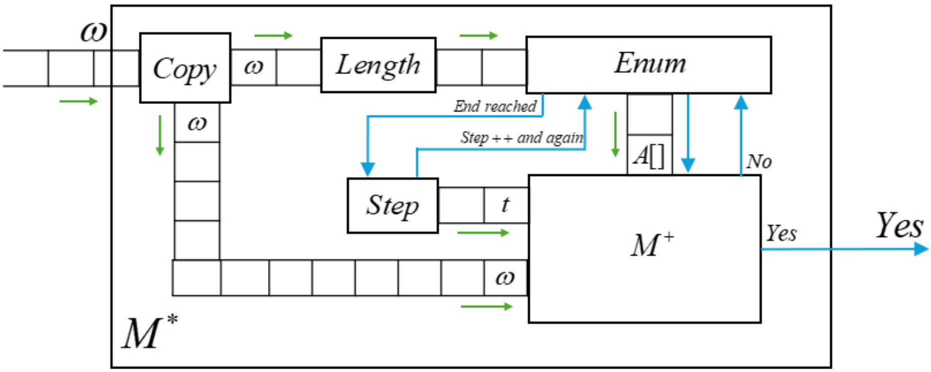
将 $TEMP$ 与 t 送入机器 M 中:

当 M 返回 Yes 时，对 $A[]$ 进行下一项的读取遍历

否则当 M 返回 No 时， M^+ 输出 No

③当 $A[]$ 遍历完毕而 M 尚未输出 No 时（即 M 所有返回值都是 Yes），M^+ 输出 Yes

最后我们构造 M^*，如图所示:



M^* 将外界输入 ω 送至 $Copy$ 中，产生两条内容与 ω 一致的输出，分别送入 $Length$ 与 M^+ 中。

$Length$ 将计算 ω 的长度，送入 $Enum$ 中。

$Enum$ 将根据输入的数值，对长度进行所有可能情况的划分（不重不漏地）并进行输出（例：对于输入 3， $Enum$ 将分别输出 $[3]$，$[1,2]$，$[2,1]$，$[1,1,1]$）。当 $Enum$ 每完成一次输出 $A[]$，就将 $A[]$ 送入 M^+ 并将控制权交给 M^+。

若 $Enum$ 穷尽了所有可能的划分且被再次调用， $Enum$ 将对自己执行“格式化”（即下次调用时从头开始进行划分的枚举），然后将控制权交给 $Step$。

$Step$ 用于计数。 $Step$ 初始值为 0。 $Step$ 始终将自己的输出作为 M^+ 的输入。每当控制权由 $Enum$ 交给 $Step$ 时， $Step$ 将自己的输出

值加1（即自增）。自增后，将控制权交回 $Enum$ 。

当 M^+ 输出 Yes 时， M^* 立即输出 Yes ；当 M^+ 输出 No 时，将控制权交给 $Enum$ 进行下一次划分的生成。

对 $\forall\omega\in L\left(M^*\right)$ ，可知对 ω 存在一个由 $Enum$ 生成的划分数组 $A[]=[a_1,a_2,...,a_s]$ ，由 $Step$ 生成的最大步数 t ，使得 M^+ 将 ω 、 $A[]$ 与 t 作为输入时，其输出为 Yes ，从而使 M^* 返回 Yes 。

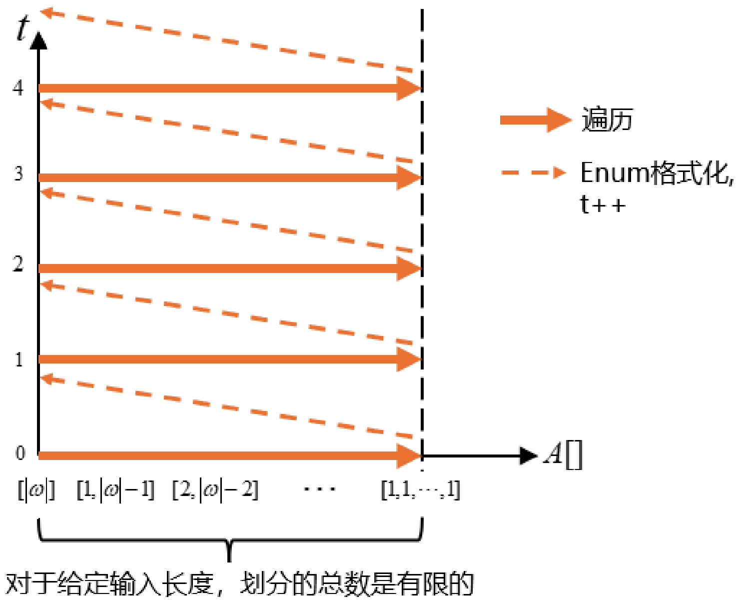
由 M^+ 的构造知，存在字符串列 $\omega_1,\omega_2,\cdots,\omega_s$ ， $\left|\omega_i\right|=a_i$ ， $\omega=\omega_1\omega_2\cdots\omega_s$ ，这些 ω_i 与 t 对 M 的输出都为 Yes ，即 $\forall i\in\left[1,s\right],\omega_i\in L\left(M\right)=L$ ，且 M 在 t 步内接受所有 ω_i ，从而 $\omega=\omega_1\omega_2\cdots\omega_s\in L^s\subseteq L^*$ ，即 $\omega\in L^*$ 。

对 $\forall\omega\in L^*$ ，知 ω 有划分 $\omega=\omega_1\omega_2\cdots\omega_s\in L^s\subseteq L^*$ ，其中 $s\geq1$ ， $\left|\omega_i\right|\geq1$ ， $\omega_i\in L$ 。从而 M' 在有限步内接收所有的 ω_i 。由 M' 机器的构造知，对每个 ω_i ，都存在 $N_i\in\mathbb{N}$ ，使得将 ω_i 与 N_i 作为 M' 的输入时， M' 返回 Yes （即对于每个 ω_i ， M 将在 N_i 步内接收 ω_i ）。

从而将 $\omega, A[]=\left[\left|\omega_1\right|,\left|\omega_2\right|,\cdots,\left|\omega_s\right|\right]$ 及 $t=\max\limits_{1\leq i\leq s}\left\{N_i\right\}$ 作为 M^+ 的输入时， M^+ 将输出 Yes 。而根据 $Enum$ 的构造， $Enum$ 总将在有限步内将上述的 $A[]$ 作为其输出， $Step$ 也将在有限步内将 t 作为其输出，这时，由上所述， M^+ 将输出 Yes ，从而 M^* 输出 Yes ，即有 $\omega\in L\left(M^*\right)$ 。

从而 $L\left(M^*\right)=L^*$，即有 M^* 识别 L^*。

（图示：$Enum$ 与 $Step$ 结合时的遍历顺序）



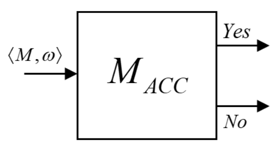
（三）不可计算证明相关的题目

题目 7：证明：L_{ACC} 语言是不可判定的（或称不是图灵可判定的）。

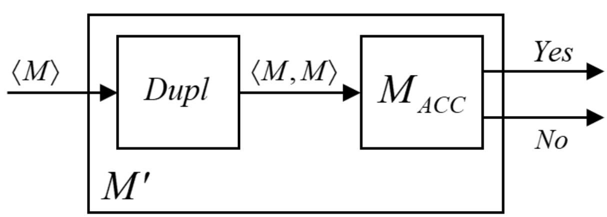
证明:

使用反证法：假设 L_{ACC} 语言是（图灵）可判定的，则存在图灵机 M_{ACC}，使得 $L(M_{ACC}) = L_{ACC}$，且 M_{ACC} 始终停机。

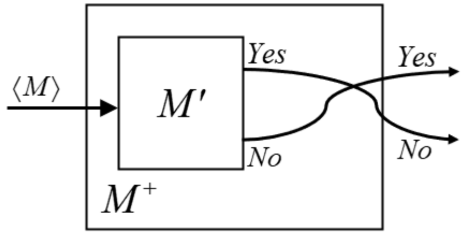
对 M_{ACC} 进行进一步描述：M_{ACC} 接收输入 $\langle M, \omega \rangle$。当 M 是一个图灵机，且 M 接收 ω 时，M_{ACC} 返回 Yes；否则返回 No。



使用 M_{ACC} 构造图灵机 M'：M' 接收输入 $\langle M \rangle$，将其复制为 $\langle M, M \rangle$ 后送入图灵机 M_{ACC} 中，最后转发 M_{ACC} 的输出为其输出。



最后使用 M' 构造图灵机 M^+：M^+ 接收输入 $\langle M \rangle$，将其转发给 M'，并反转 M' 的输出后输出。



现考察，当将$\langle M^+\rangle$作为M^+的输入时，M^+的输出结果：

①当$M^+\left(\langle M^+\rangle\right)=Yes$时，由$M^+$构造，知$M'\left(\langle M^+\rangle\right)=No$，由$M'$构造，知$M_{ACC}\left(\langle M^+,M^+\rangle\right)=No$，从而$M^+$不接收$\langle M^+\rangle$，故$M^+\left(\langle M^+\rangle\right)=No$，矛盾；

②当$M^+\left(\langle M^+\rangle\right)=No$时，由$M^+$构造，知$M'\left(\langle M^+\rangle\right)=Yes$，由$M'$构造，知$M_{ACC}\left(\langle M^+,M^+\rangle\right)=Yes$，从而$M^+$接收$\langle M^+\rangle$，故$M^+\left(\langle M^+\rangle\right)=Yes$，矛盾。

从而M^+这个机器根本不存在，但是在假设M_{ACC}存在的情况下，M'与M^+的构造都是合理且真实存在的，从而只能是M_{ACC}不存在，故L_{ACC}是不可（图灵）判定的。

题目8： 假定用课程讲授的图灵机编码，证明如下语言不可识别。

（1）$L = \{w_i | w_i \notin L(M_{2i})\}$

证明:

对于课程所讲授的图灵机编码：每个图灵机编码的最后一定是一条转移规则，每一条转移规则被编码为$0^{i+1}10^j10^{k+1}10^l10^m$，而$m=\begin{cases}1 & D=L\\ 2 & D=R\end{cases}$，从而图灵机编码的最后一位一定是$0$，从而每个合法的图灵机编码都一定是一个偶数。

使用对角线法证明：

$$
\begin{array}{c|cccccc}
 & \omega_0 & \omega_1 & \cdots & \omega_i & \cdots \\
\hline
M_0 & a_{00} & a_{01} & \cdots & a_{0i} & \cdots \\
M_1 & a_{10} & a_{11} & \cdots & a_{1i} & \cdots \\
M_2 & a_{20} & a_{21} & \cdots & a_{2i} & \cdots \\
M_3 & a_{30} & a_{31} & \cdots & a_{3i} & \cdots \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
M_{2i} & a_{2i,0} & a_{2i,1} & \cdots & a_{2i,i} & \cdots \\
M_{2i+1} & a_{2i+1,0} & a_{2i+1,1} & \cdots & a_{2i+1,i} & \cdots \\
\vdots & \vdots & \vdots & \cdots & \vdots & \ddots
\end{array}
$$

取M_L为$\left[\overline{a_{00}},\overline{a_{21}},\cdots,\overline{a_{2i,i}},\cdots\right]$：

首先M_L不可能是任何图灵机M_{2i}，因为$M_L[i] \neq M_{2i}[i]$。并且M_L的编码也不可能和任何M_{2i+1}相同（因为这时M_L不是有效的图灵机），从而这样的M_L根本不存在。故不存在图灵机识别语言L。

（2）$L = \{w_j | w_{2j} \notin L(M_j)\}$

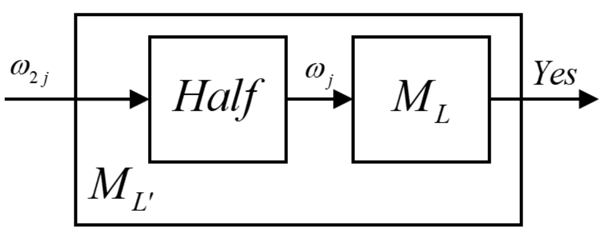
证明:

先证：语言 $L' = \left\{\omega_{2j} \mid \omega_{2j} \notin L\left(M_j\right)\right\}$ 不是递归可枚举的，对角线法:

$$
\begin{array}{c|cccccccc}
 & \omega_0 & \omega_1 & \omega_2 & \omega_3 & \cdots & \omega_{2i} & \omega_{2i+1} & \cdots \\
\hline
M_0 & a_{00} & a_{01} & a_{02} & a_{03} & \cdots & a_{0,2i} & a_{0,2i+1} & \cdots \\
M_1 & a_{10} & a_{11} & a_{12} & a_{13} & \cdots & a_{1,2i} & a_{1,2i+1} & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
M_i & a_{i0} & a_{i1} & a_{i2} & a_{i3} & \cdots & a_{i,2i} & a_{i,2i+1} & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \ddots
\end{array}
$$

取 $M_{L'}$ 为 $\left[\overline{a_{01}}, 0, \overline{a_{12}}, 0, \cdots, \overline{a_{i,2i}}, 0, \cdots\right]$，则 $M_{L'}$ 不存在，从而 $L' = \left\{\omega_{2j} \mid \omega_{2j} \notin L\left(M_j\right)\right\}$ 不是递归可枚举的。

反证：若 $L = \left\{\omega_j \mid \omega_{2j} \notin L\left(M_j\right)\right\}$ 是递归可枚举的，则存在图灵机 M_L 识别 L。现构造图灵机 $M_{L'}$：



其中 $Half$ 将 ω_{2j} 转换为 ω_j，易见 $L\left(M_{L'}\right) = L'$，矛盾。从而 L 非递归可枚举。

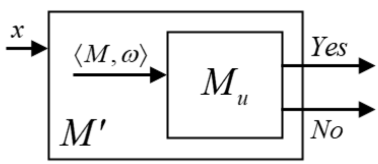
题目 9.0：证明：L_{ACC01} 语言是不可判定的。

证明:

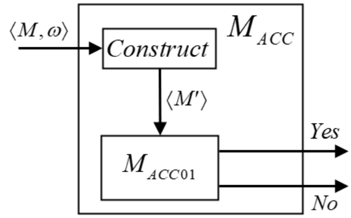
使用反证法：假设 L_{ACC01} 语言是（图灵）可判定的，则存在图灵机 M_{ACC01}，使得 $L(M_{ACC01}) = L_{ACC01}$，且 M_{ACC01} 始终停机。

对 M_{ACC01} 进行进一步描述：M_{ACC01} 接收输入 $\langle M \rangle$。当 M 是一个图灵机，且 M 接收字符串 01 时，M_{ACC01} 返回 Yes；否则返回 No。

构造图灵机 M'：M' 接收输入字符串 x，并舍弃它。在其内部模拟常量字符串 ω 在常量图灵机 M 上的运行，并转发输出其结果。



最后构造图灵机 M_{ACC}：M_{ACC} 接收输入 $\langle M, \omega \rangle$，通过内部逻辑，构造出一个以常量形式的图灵机 M，常量形式的字符串 ω 为其内部常量的图灵机 M' 的编码 $\langle M' \rangle$，之后将 $\langle M' \rangle$ 送入 M_{ACC01}，并转发输出其结果。



现证明对于这样的M_{ACC}，M_{ACC}判定语言L_{ACC}：

对$\forall\langle M,\omega\rangle\in L(M_{ACC})$，有$M_{ACC}(\langle M,\omega\rangle)=Yes$，从而$M_{ACC01}(\langle M'\rangle)=Yes$，从而$M'$接收字符串$01$。但$M'$的输出与其输入$x$无关，当$M'$接收$01$时，$M'$就接收了所有的输入$x\in\Sigma^*$，即$M'$输出常量$Yes$，从而对于常量图灵机$M$，常量字符串$\omega$，$M_u(\langle M,\omega\rangle)$返回常量$Yes$，从而$M$接收$\omega$，$\langle M,\omega\rangle\in L_{ACC}$。

对$\forall\langle M,\omega\rangle\notin L(M_{ACC})$，有$M_{ACC}(\langle M,\omega\rangle)=No$，从而$M_{ACC01}(\langle M'\rangle)=No$，从而$M'$不接收字符串$01$。但$M'$的输出与其输入$x$无关，当$M'$不接收$01$时，$M'$就不接收所有的输入$x\in\Sigma^*$，即$M'$输出常量$No$，从而对于常量图灵机$M$，常量字符串$\omega$，$M_u(\langle M,\omega\rangle)$返回常量$No$，从而$M$不接收$\omega$，$\langle M,\omega\rangle\notin L_{ACC}$。

从而$L(M_{ACC})=L_{ACC}$，且无论是$Construct$构造M'的过程，还是M_{ACC}的运行，都是始终停机的，从而M_{ACC}始终停机，故M_{ACC}判定L_{ACC}。

但根据定理，L_{ACC}是不可判定的，矛盾。从而M_{ACC01}不存在，从而L_{ACC01}是不可（图灵）判定的。