

Algoritmuselmélet 13. előadás

Katona Gyula Y.
Budapesti Műszaki és Gazdaságtudományi Egyetem
Számítástudományi Tsz.
I. B. 137/b
kiskat@cs.bme.hu

2002 Április 15.

Univerzális Turing gép

Turing-gép \leftrightarrow program

Univerzális Turing gép

Turing-gép \leftrightarrow program

Univerzális Turing-gép \leftrightarrow fordító program (interpreter)

Univerzális Turing gép

Turing-gép \leftrightarrow program

Univerzális Turing-gép \leftrightarrow fordító program (interpreter)

M TG leírása és $s \in I^*$ bemenete \implies Univerzális TG \implies szimulálja M -et
 s bemenettel

Univerzális Turing gép

Turing-gép \leftrightarrow program

Univerzális Turing-gép \leftrightarrow fordító program (interpreter)

M TG leírása és $s \in I^*$ bemenete \implies Univerzális TG \implies szimulálja M -et s bemenettel

Hogyan írjuk le az M TG-et?

Univerzális Turing gép

Turing-gép \leftrightarrow program

Univerzális Turing-gép \leftrightarrow fordító program (interpreter)

M TG leírása és $s \in I^*$ bemenete \implies Univerzális TG \implies szimulálja M -et s bemenettel

Hogyan írjuk le az M TG-et?

Tegyük fel, hogy $k = 1$, $M = (Q, T, I, \ddot{u}, \delta, q_0, F)$, $I = \{0, 1\}$ és $|F| = 1$.
Pl.

- $I = \{0, 1\}$, $T = \{0, 1, \dots, t\}$, $\ddot{u} = t$,
- $Q = \{0, 1, \dots, q\}$, $q_0 = 0$, $F = \{q\}$,
- balra = 0, jobbra = 1, helyben = 2

Ekkor az M Turing-gép leírása, kódja \implies

$$q\#t\#q_1\#x_1\#q'_1\#x'_1\#m'_1\#\dots\#q_r\#x_r\#q'_r\#x'_r\#m'_r\#\#,$$

ahol a megfelelő számokat binárisan írjuk le, továbbá δ értékeit a következőképpen soroljuk fel (ahol értelmezett):

$$\text{a } \delta(q_i, x_i) = (q'_i, x'_i, m'_i) \text{ tény kódja } q_i\#x_i\#q'_i\#x'_i\#m'_i.$$

Ekkor az M Turing-gép leírása, kódja \implies

$$q\#t\#q_1\#x_1\#q'_1\#x'_1\#m'_1\#\dots\#q_r\#x_r\#q'_r\#x'_r\#m'_r\#\#,$$

ahol a megfelelő számokat binárisan írjuk le, továbbá δ értékeit a következőképpen soroljuk fel (ahol értelmezett):

$$\text{a } \delta(q_i, x_i) = (q'_i, x'_i, m'_i) \text{ tény kódja } q_i\#x_i\#q'_i\#x'_i\#m'_i.$$

Minden szóba jövő M gépet egy $w \in I^*$ szóval írunk le.

Ekkor az M Turing-gép leírása, kódja \implies

$$q\#t\#q_1\#x_1\#q'_1\#x'_1\#m'_1\#\dots\#q_r\#x_r\#q'_r\#x'_r\#m'_r\#\#,$$

ahol a megfelelő számokat binárisan írjuk le, továbbá δ értékeit a következőképpen soroljuk fel (ahol értelmezett):

$$\text{a } \delta(q_i, x_i) = (q'_i, x'_i, m'_i) \text{ tény kódja } q_i\#x_i\#q'_i\#x'_i\#m'_i.$$

Minden szóba jövő M gépet egy $w \in I^*$ szóval írunk le.

Tetszőleges $w \in I^*$ szóhoz legfeljebb egy gép van, amelynek a kódja $w \implies M_w$

Ekkor az M Turing-gép leírása, kódja \implies

$$q\#t\#q_1\#x_1\#q'_1\#x'_1\#m'_1\#\dots\#q_r\#x_r\#q'_r\#x'_r\#m'_r\#\#,$$

ahol a megfelelő számokat binárisan írjuk le, továbbá δ értékeit a következőképpen soroljuk fel (ahol értelmezett):

$$\text{a } \delta(q_i, x_i) = (q'_i, x'_i, m'_i) \text{ tény kódja } q_i\#x_i\#q'_i\#x'_i\#m'_i.$$

Minden szóba jövő M gépet egy $w \in I^*$ szóval írunk le.

Tetszőleges $w \in I^*$ szóhoz legfeljebb egy gép van, amelynek a kódja

$$w \implies M_w$$

Egymásból kiszámolható M és M_w .

Tétel. *Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).*

Tétel. *Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).*

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Tétel. Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Második szalag $\implies M_w$ egyetlen szalagjának felel meg.

Tétel. Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Második szalag $\implies M_w$ egyetlen szalagjának felel meg.

Harmadik szalag \implies az M_w belső állapota

Tétel. Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Második szalag $\implies M_w$ egyetlen szalagjának felel meg.

Harmadik szalag \implies az M_w belső állapota

Előkészítés \implies ellenőrzi, hogy M_w létezik-e.

Tétel. Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Második szalag $\implies M_w$ egyetlen szalagjának felel meg.

Harmadik szalag \implies az M_w belső állapota

Előkészítés \implies ellenőrzi, hogy M_w létezik-e.

\implies NEM \rightarrow megáll elutasító állapotban

Tétel. Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Második szalag $\implies M_w$ egyetlen szalagjának felel meg.

Harmadik szalag \implies az M_w belső állapota

Előkészítés \implies ellenőrzi, hogy M_w létezik-e.

\implies NEM \rightarrow megáll elutasító állapotban

\implies IGEN \rightarrow átmásolja az s inputot a második szalagjára, és a harmadik szalagra a kezdőállapot kódját jegyzi fel.

Tétel. Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Második szalag $\implies M_w$ egyetlen szalagjának felel meg.

Harmadik szalag \implies az M_w belső állapota

Előkészítés \implies ellenőrzi, hogy M_w létezik-e.

\implies NEM \rightarrow megáll elutasító állapotban

\implies IGEN \rightarrow átmásolja az s inputot a második szalagjára, és a harmadik szalagra a kezdőállapot kódját jegyzi fel.

$w\#s$
s
q_0

Tétel. Van olyan 3-szalagos U Turing-gép, amelyre teljesül a következő: ha $w, s \in I^*$, és M_w létezik, akkor az U gép a $w\#s$ bemenetet pontosan akkor fogadja el (utasítja el, kerül vele végtelen ciklusba), ha M_w az s bemenetet elfogadja (elutasítja, végtelen ciklusba kerül vele).

Bizonyítás: (vázlat)

Első szalag $\implies w\#s$ input, w értelmezgetése

Második szalag $\implies M_w$ egyetlen szalagjának felel meg.

Harmadik szalag \implies az M_w belső állapota

Előkészítés \implies ellenőrzi, hogy M_w létezik-e.

\implies **NEM** \rightarrow megáll elutasító állapotban

\implies **IGEN** \rightarrow átmásolja az s inputot a második szalagjára, és a harmadik szalagra a kezdőállapot kódját jegyzi fel.

$w\#s$
s
q_0

Az U az M_w gép egy lépését több lépésben szimulálja \implies

$w \# s$

M_w szalagja az i -edik lépés után
--

M_w belső állapota az i -edik lépés után
--

$w \# s$
M_w szalagja az i -edik lépés után
M_w belső állapota az i -edik lépés után

U akkor áll meg, ha M_w megáll,

$w\#s$
M_w szalagja az i -edik lépés után
M_w belső állapota az i -edik lépés után

U akkor áll meg, ha M_w megáll, pontosan akkor fogadja el a $w\#s$ bemenetét, ha a megállás után az M_w elfogadó állapotának kódja van az utolsó szalagon. ✓

Alapvető kiszámíthatatlansági tételek

Be fogjuk látni, hogy

$$\mathcal{R} \subsetneq \mathcal{RE} \subsetneq 2^{I^*}.$$

Alapvető kiszámíthatatlansági tételek

Be fogjuk látni, hogy

$$\mathcal{R} \subsetneq \mathcal{RE} \subsetneq 2^{I^*}.$$

Definíció. Azon gépek kódjainak L_d nyelve, amik nem fogadják el saját kódjukat a *diagonális nyelv*:

$$L_d = \{w \in I^*; \text{ az } M_w \text{ gép létezik, és } w \notin L_{M_w}\}.$$

Alapvető kiszámíthatatlansági tételek

Be fogjuk látni, hogy

$$\mathcal{R} \subsetneq \mathcal{RE} \subsetneq 2^{I^*}.$$

Definíció. Azon gépek kódjainak L_d nyelve, amik nem fogadják el saját kódjukat a *diagonális nyelv*:

$$L_d = \{w \in I^*; \text{ az } M_w \text{ gép létezik, és } w \notin L_{M_w}\}.$$

Tétel. L_d nem rekurzíve felsorolható.

Alapvető kiszámíthatatlansági tételek

Be fogjuk látni, hogy

$$\mathcal{R} \subsetneq \mathcal{RE} \subsetneq 2^{I^*}.$$

Definíció. Azon gépek kódjainak L_d nyelve, amik nem fogadják el saját kódjukat a *diagonális nyelv*:

$$L_d = \{w \in I^*; \text{ az } M_w \text{ gép létezik, és } w \notin L_{M_w}\}.$$

Tétel. L_d nem rekurzíve felsorolható.

Bizonyítás: Indirekt, tegyük fel hogy rekurzíve felsorolható $\implies \exists M$ TG amire $L_d = L_M$, ennek kódja legyen w

Alapvető kiszámíthatatlansági tételek

Be fogjuk látni, hogy

$$\mathcal{R} \subsetneq \mathcal{RE} \subsetneq 2^{I^*}.$$

Definíció. Azon gépek kódjainak L_d nyelve, amik nem fogadják el saját kódjukat a *diagonális nyelv*:

$$L_d = \{w \in I^*; \text{ az } M_w \text{ gép létezik, és } w \notin L_{M_w}\}.$$

Tétel. L_d nem rekurzíve felsorolható.

Bizonyítás: Indirekt, tegyük fel hogy rekurzíve felsorolható $\implies \exists M$ TG amire $L_d = L_M$, ennek kódja legyen w

- $w \in L_d \implies L_d$ definíciója szerint $w \notin L_{M_w} = L_d$

Alapvető kiszámíthatatlansági tételek

Be fogjuk látni, hogy

$$\mathcal{R} \subsetneq \mathcal{RE} \subsetneq 2^{I^*}.$$

Definíció. Azon gépek kódjainak L_d nyelve, amik nem fogadják el saját kódjukat a *diagonális nyelv*:

$$L_d = \{w \in I^*; \text{ az } M_w \text{ gép létezik, és } w \notin L_{M_w}\}.$$

Tétel. L_d nem rekurzíve felsorolható.

Bizonyítás: Indirekt, tegyük fel hogy rekurzíve felsorolható $\implies \exists M$ TG amire $L_d = L_M$, ennek kódja legyen w

- $w \in L_d \implies L_d$ definíciója szerint $w \notin L_{M_w} = L_d$ ⚡
- $w \notin L_d \implies L_d$ definíciója szerint $w \in L_d = L_{M_w}$ ⚡

Az univerzális nyelv

Definíció. Az olyan (TG kód, input szó) párok L_u nyelve, amelyekre a gép elfogadja az inputot az *univerzális nyelv*:

$$L_u = \{w\#s \in I^*; \text{ az } M_w \text{ gép létezik, és } s \in L_{M_w}\}.$$

Az univerzális nyelv

Definíció. Az olyan (TG kód, input szó) párok L_u nyelve, amelyekre a gép elfogadja az inputot az *univerzális nyelv*:

$$L_u = \{w\#s \in I^*; \text{ az } M_w \text{ gép létezik, és } s \in L_{M_w}\}.$$

Tétel. [A. Turing, 1936] L_u egy rekurzíve felsorolható, de nem rekurzív nyelv.

Az univerzális nyelv

Definíció. Az olyan (TG kód, input szó) párok L_u nyelve, amelyekre a gép elfogadja az inputot az *univerzális nyelv*:

$$L_u = \{w\#s \in I^*; \text{ az } M_w \text{ gép létezik, és } s \in L_{M_w}\}.$$

Tétel. [A. Turing, 1936] L_u egy rekurzíve felsorolható, de nem rekurzív nyelv.

Bizonyítás: L_u -t éppen az univerzális Turing-gépek ismerik fel $\implies L_u$ rekurzíve felsorolható.

Az univerzális nyelv

Definíció. Az olyan (TG kód, input szó) párok L_u nyelve, amelyekre a gép elfogadja az inputot az *univerzális nyelv*:

$$L_u = \{w\#s \in I^*; \text{ az } M_w \text{ gép létezik, és } s \in L_{M_w}\}.$$

Tétel. [A. Turing, 1936] L_u egy rekurzíve felsorolható, de nem rekurzív nyelv.

Bizonyítás: L_u -t éppen az univerzális Turing-gépek ismerik fel $\implies L_u$ rekurzíve felsorolható.

Tegyük fel indirekt, hogy L_u rekurzív; legyen M egy mindig megálló TG, ami felismeri L_u -t.

Az univerzális nyelv

Definíció. Az olyan (TG kód, input szó) párok L_u nyelve, amelyekre a gép elfogadja az inputot az *univerzális nyelv*:

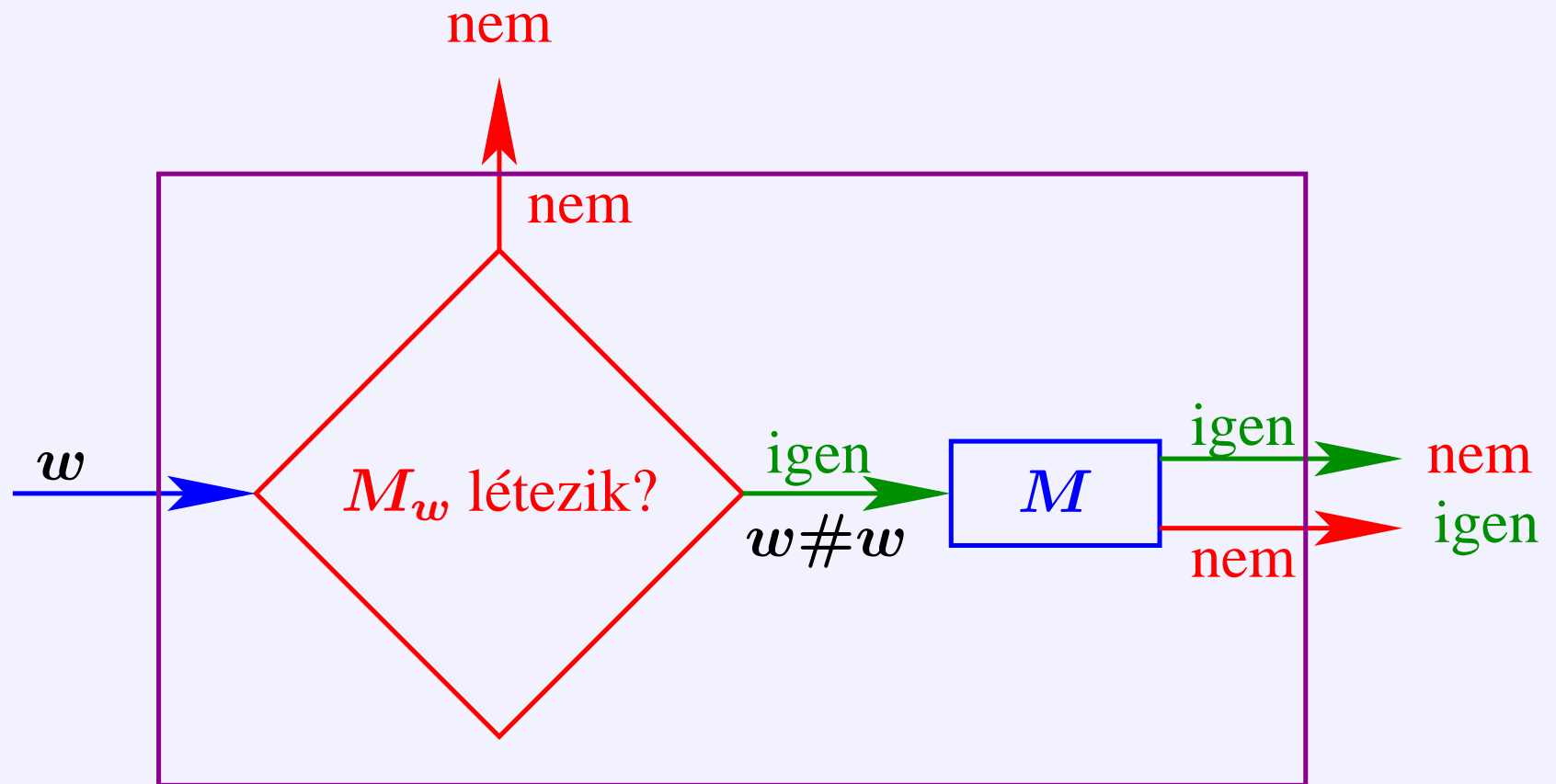
$$L_u = \{w\#s \in I^*; \text{ az } M_w \text{ gép létezik, és } s \in L_{M_w}\}.$$

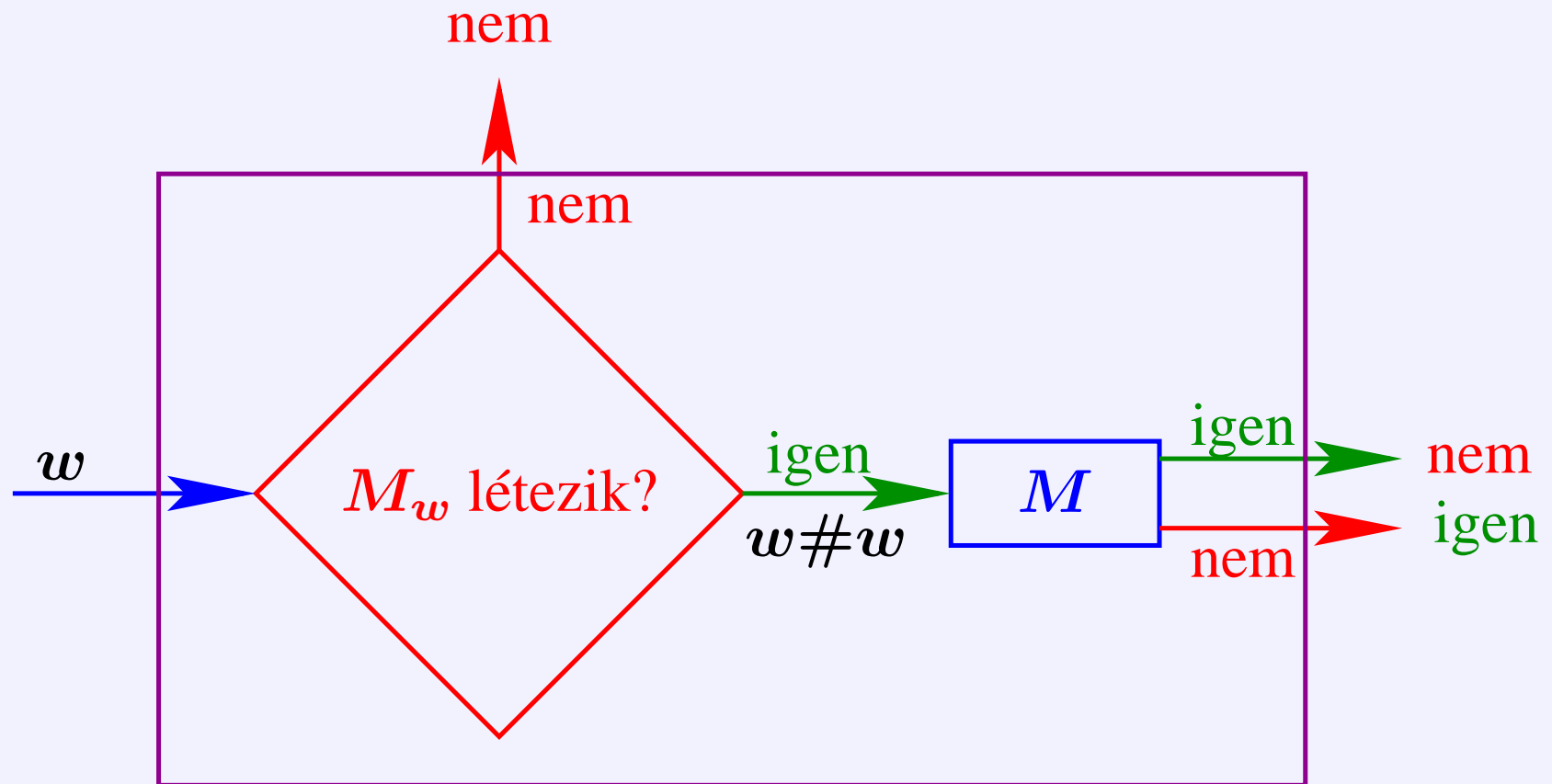
Tétel. [A. Turing, 1936] L_u egy rekurzíve felsorolható, de nem rekurzív nyelv.

Bizonyítás: L_u -t éppen az univerzális Turing-gépek ismerik fel $\implies L_u$ rekurzíve felsorolható.

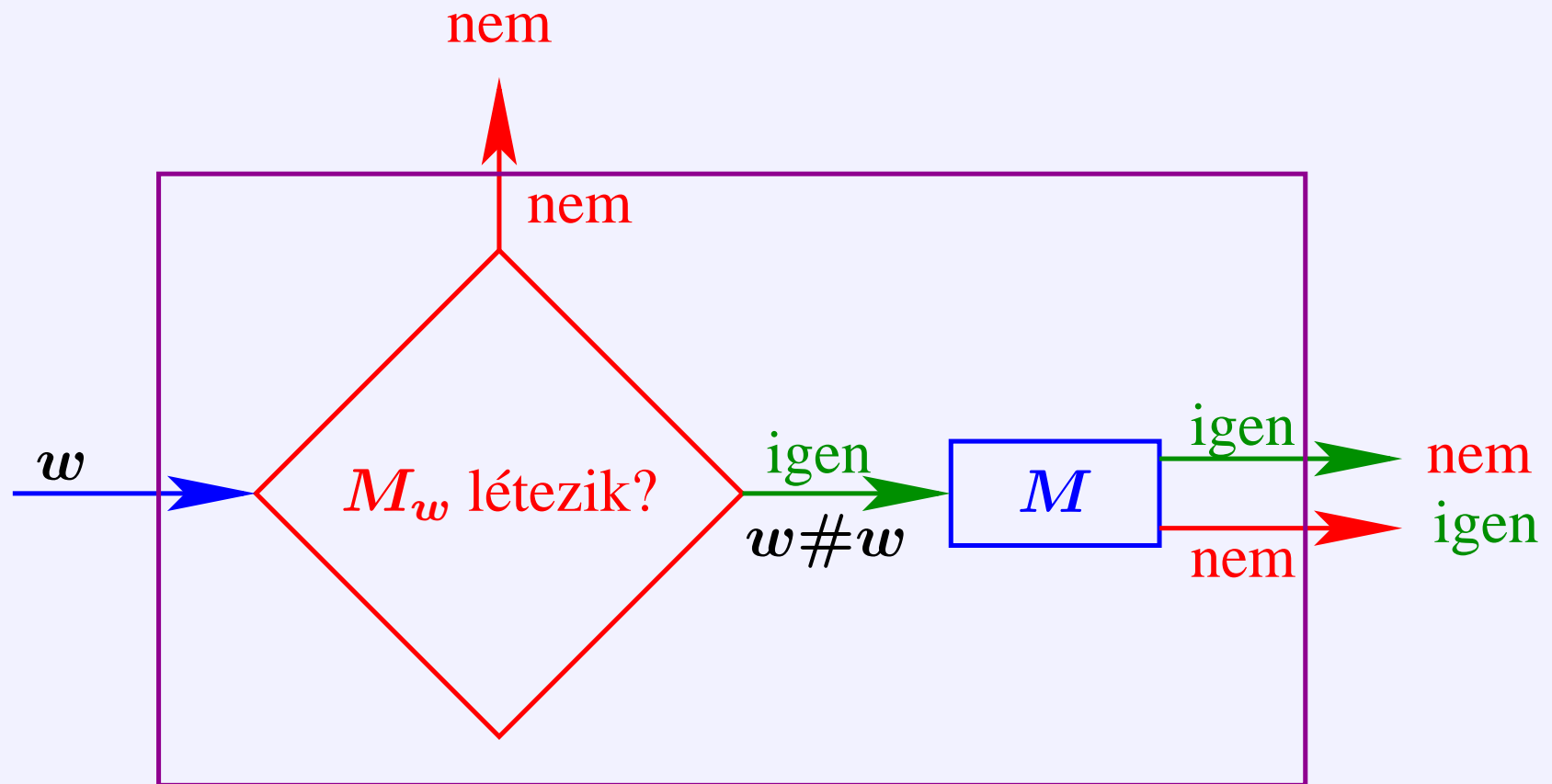
Tegyük fel indirekt, hogy L_u rekurzív; legyen M egy mindig megálló TG, ami felismeri L_u -t.

Konstruáljuk meg az M' TG-et:



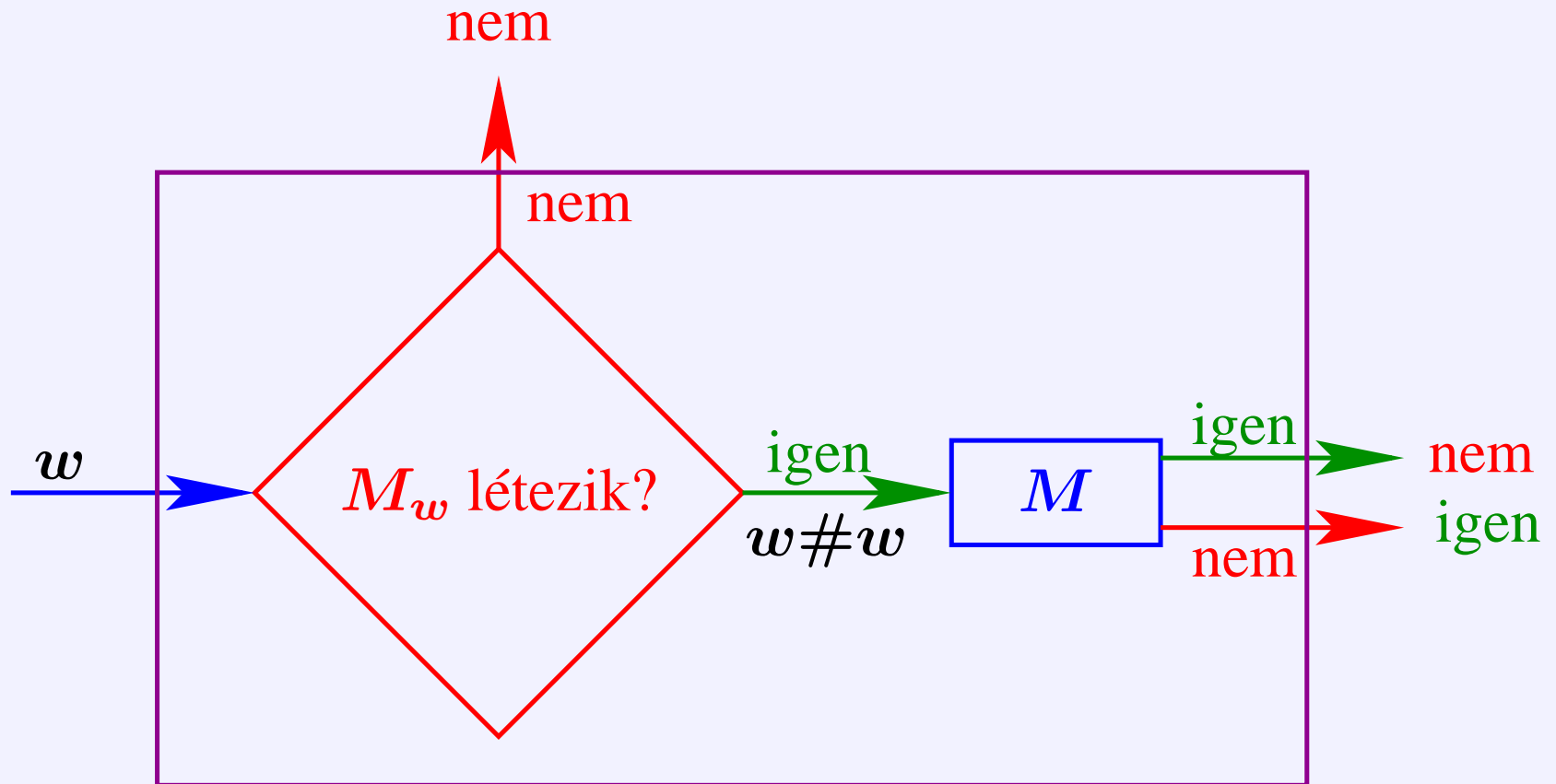


Az M' gép mindig megáll, mert M mindig megáll.



Az M' gép mindig megáll, mert M mindig megáll.

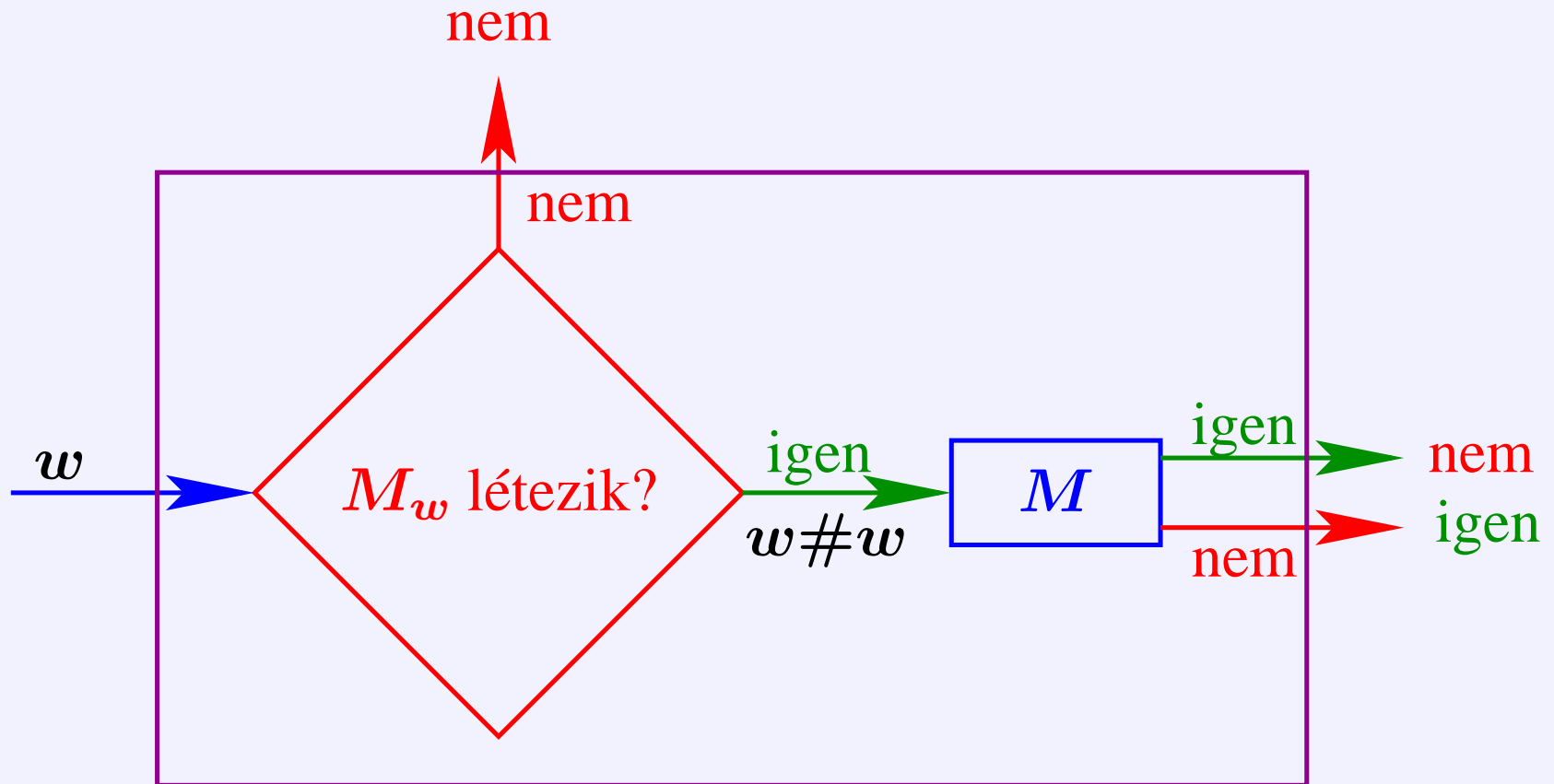
M' pontosan akkor fogadja el w -t, ha M_w létezik és $w \notin L_{M_w}$.



Az M' gép mindig megáll, mert M mindig megáll.

M' pontosan akkor fogadja el w -t, ha M_w létezik és $w \notin L_{M_w}$.

$\implies M'$ éppen az L_d nyelvet fogadja el



Az M' gép mindig megáll, mert M mindig megáll.

M' pontosan akkor fogadja el w -t, ha M_w létezik és $w \notin L_{M_w}$.

$\implies M'$ éppen az L_d nyelvet fogadja el \checkmark
 hiszen L_d nem rekurzíve felsorolható \checkmark

Összefüggések a kiszámíthatósági fogalmak között

Ha van egy mindig megálló M algoritmusunk L felismerésére, akkor van algoritmus az $I^* \setminus L$ nyelv felismerésére is.

Összefüggések a kiszámíthatósági fogalmak között

Ha van egy mindig megálló M algoritmusunk L felismerésére, akkor van algoritmus az $I^* \setminus L$ nyelv felismerésére is.

Ha csak olyan „fél” algoritmusunk van, ami nem mindig áll meg, ezt nem lehet megtenni.

Összefüggések a kiszámíthatósági fogalmak között

Ha van egy mindig megálló M algoritmusunk L felismerésére, akkor van algoritmus az $I^* \setminus L$ nyelv felismerésére is.

Ha csak olyan „fél” algoritmusunk van, ami nem mindig áll meg, ezt nem lehet megtenni.

Ha van fél algoritmus L -re és $I^* \setminus L$ -re is, akkor ebből összejön egy egész algoritmus

Összefüggések a kiszámíthatósági fogalmak között

Ha van egy mindig megálló M algoritmusunk L felismerésére, akkor van algoritmus az $I^* \setminus L$ nyelv felismerésére is.

Ha csak olyan „fél” algoritmusunk van, ami nem mindig áll meg, ezt nem lehet megtenni.

Ha van fél algoritmus L -re és $I^* \setminus L$ -re is, akkor ebből összejön egy egész algoritmus

Definíció. A nyelvekből álló halmazokat (2^{I^*} részhalmazait) **nyelvosztályoknak** nevezzük. (Pl. \mathcal{R} , \mathcal{RE} , 2^{I^*} .)

Összefüggések a kiszámíthatósági fogalmak között

Ha van egy mindig megálló M algoritmusunk L felismerésére, akkor van algoritmus az $I^* \setminus L$ nyelv felismerésére is.

Ha csak olyan „fél” algoritmusunk van, ami nem mindig áll meg, ezt nem lehet megtenni.

Ha van fél algoritmus L -re és $I^* \setminus L$ -re is, akkor ebből összejön egy egész algoritmus

Definíció. A nyelvekből álló halmazokat (2^{I^*} részhalmazait) **nyelvosztályoknak** nevezzük. (Pl. \mathcal{R} , \mathcal{RE} , 2^{I^*} .)

Legyen $X \subseteq 2^{I^*}$ egy nyelv osztály. Ekkor a **komplementer nyelv osztály**, $\text{co}X$ az X -beli nyelvek komplementereiből áll:

$$\text{co}X = \{L \subseteq I^* : I^* \setminus L \in X\}.$$

Összefüggések a kiszámíthatósági fogalmak között

Ha van egy mindig megálló M algoritmusunk L felismerésére, akkor van algoritmus az $I^* \setminus L$ nyelv felismerésére is.

Ha csak olyan „fél” algoritmusunk van, ami nem mindig áll meg, ezt nem lehet megtenni.

Ha van fél algoritmus L -re és $I^* \setminus L$ -re is, akkor ebből összejön egy egész algoritmus

Definíció. A nyelvekből álló halmazokat (2^{I^*} részhalmazait) **nyelvosztályoknak** nevezzük. (Pl. \mathcal{R} , \mathcal{RE} , 2^{I^*} .)

Legyen $X \subseteq 2^{I^*}$ egy nyelv osztály. Ekkor a **komplementer nyelv osztály**, $\text{co}X$ az X -beli nyelvek komplementereiből áll:

$$\text{co}X = \{L \subseteq I^* : I^* \setminus L \in X\}.$$

\implies

$$X \subseteq Y \subseteq 2^{I^*} \implies \text{co}X \subseteq \text{co}Y.$$

Összefüggések a kiszámíthatósági fogalmak között

Ha van egy mindig megálló M algoritmusunk L felismerésére, akkor van algoritmus az $I^* \setminus L$ nyelv felismerésére is.

Ha csak olyan „fél” algoritmusunk van, ami nem mindig áll meg, ezt nem lehet megtenni.

Ha van fél algoritmus L -re és $I^* \setminus L$ -re is, akkor ebből összejön egy egész algoritmus

Definíció. A nyelvekből álló halmazokat (2^{I^*} részhalmazait) **nyelvosztályoknak** nevezzük. (Pl. \mathcal{R} , \mathcal{RE} , 2^{I^*} .)

Legyen $X \subseteq 2^{I^*}$ egy nyelv osztály. Ekkor a **komplementer nyelv osztály**, $\text{co}X$ az X -beli nyelvek komplementereiből áll:

$$\text{co}X = \{L \subseteq I^* : I^* \setminus L \in X\}.$$

\implies

$$X \subseteq Y \subseteq 2^{I^*} \implies \text{co}X \subseteq \text{co}Y.$$

$$\text{co}(\text{co}X) = X$$

Tétel. $\mathcal{R} = \text{co}\mathcal{R}$

Bizonyítás: Ha $L \in \mathcal{R} \implies \exists M$ TG, mely minden inputon megáll és az L nyelvet fogadja el.

Tétel. $\mathcal{R} = \text{co}\mathcal{R}$

Bizonyítás: Ha $L \in \mathcal{R} \implies \exists M$ TG, mely minden inputon megáll és az L nyelvet fogadja el.

Cseréljük fel M elfogadó és elutasítva megálló állapotait $\implies \text{co}\mathcal{R} \subseteq \mathcal{R}$. ✓

Tétel. $\mathcal{R} = \text{co}\mathcal{R}$

Bizonyítás: Ha $L \in \mathcal{R} \implies \exists M$ TG, mely minden inputon megáll és az L nyelvet fogadja el.

Cseréljük fel M elfogadó és elutasítva megálló állapotait $\implies \text{co}\mathcal{R} \subseteq \mathcal{R}$. ✓

Másik irány:

$$\mathcal{R} = \text{co}(\text{co}\mathcal{R}) \subseteq \text{co}\mathcal{R}. \quad \checkmark$$

Tétel. $\mathcal{R} = \text{co}\mathcal{R}$

Bizonyítás: Ha $L \in \mathcal{R} \implies \exists M \text{ TG}$, mely minden inputon megáll és az L nyelvet fogadja el.

Cseréljük fel M elfogadó és elutasítva megálló állapotait $\implies \text{co}\mathcal{R} \subseteq \mathcal{R}$. ✓

Másik irány:

$$\mathcal{R} = \text{co}(\text{co}\mathcal{R}) \subseteq \text{co}\mathcal{R}. \quad \checkmark$$

Tétel. $\mathcal{R} = \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E}$

Bizonyítás: $\mathcal{R} \subseteq \mathcal{R}\mathcal{E} \implies \mathcal{R} = \text{co}\mathcal{R} \subseteq \text{co}\mathcal{R}\mathcal{E} \implies \mathcal{R} \subseteq \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E}$

Tétel. $\mathcal{R} = \text{co}\mathcal{R}$

Bizonyítás: Ha $L \in \mathcal{R} \implies \exists M$ TG, mely minden inputon megáll és az L nyelvet fogadja el.

Cseréljük fel M elfogadó és elutasítva megálló állapotait $\implies \text{co}\mathcal{R} \subseteq \mathcal{R}$. ✓

Másik irány:

$$\mathcal{R} = \text{co}(\text{co}\mathcal{R}) \subseteq \text{co}\mathcal{R}. \quad \checkmark$$

Tétel. $\mathcal{R} = \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E}$

Bizonyítás: $\mathcal{R} \subseteq \mathcal{R}\mathcal{E} \implies \mathcal{R} = \text{co}\mathcal{R} \subseteq \text{co}\mathcal{R}\mathcal{E} \implies \mathcal{R} \subseteq \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E}$

Másik irány:

Tegyük fel, hogy $L \in \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E} \implies$ legyen M_1 , illetve M_2 két TG, melyek az L , illetve az $I^* \setminus L$ nyelvet ismerik fel.

Tétel. $\mathcal{R} = \text{co}\mathcal{R}$

Bizonyítás: Ha $L \in \mathcal{R} \implies \exists M$ TG, mely minden inputon megáll és az L nyelvet fogadja el.

Cseréljük fel M elfogadó és elutasítva megálló állapotait $\implies \text{co}\mathcal{R} \subseteq \mathcal{R}$. ✓

Másik irány:

$$\mathcal{R} = \text{co}(\text{co}\mathcal{R}) \subseteq \text{co}\mathcal{R}. \quad \checkmark$$

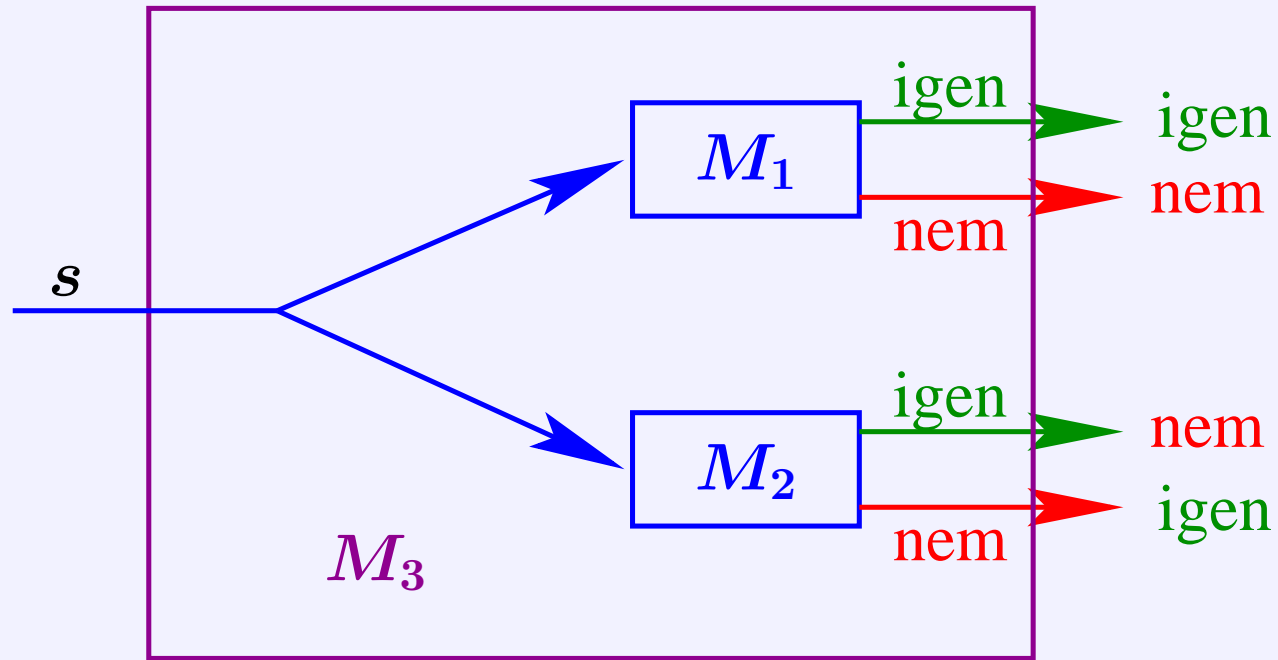
Tétel. $\mathcal{R} = \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E}$

Bizonyítás: $\mathcal{R} \subseteq \mathcal{R}\mathcal{E} \implies \mathcal{R} = \text{co}\mathcal{R} \subseteq \text{co}\mathcal{R}\mathcal{E} \implies \mathcal{R} \subseteq \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E}$

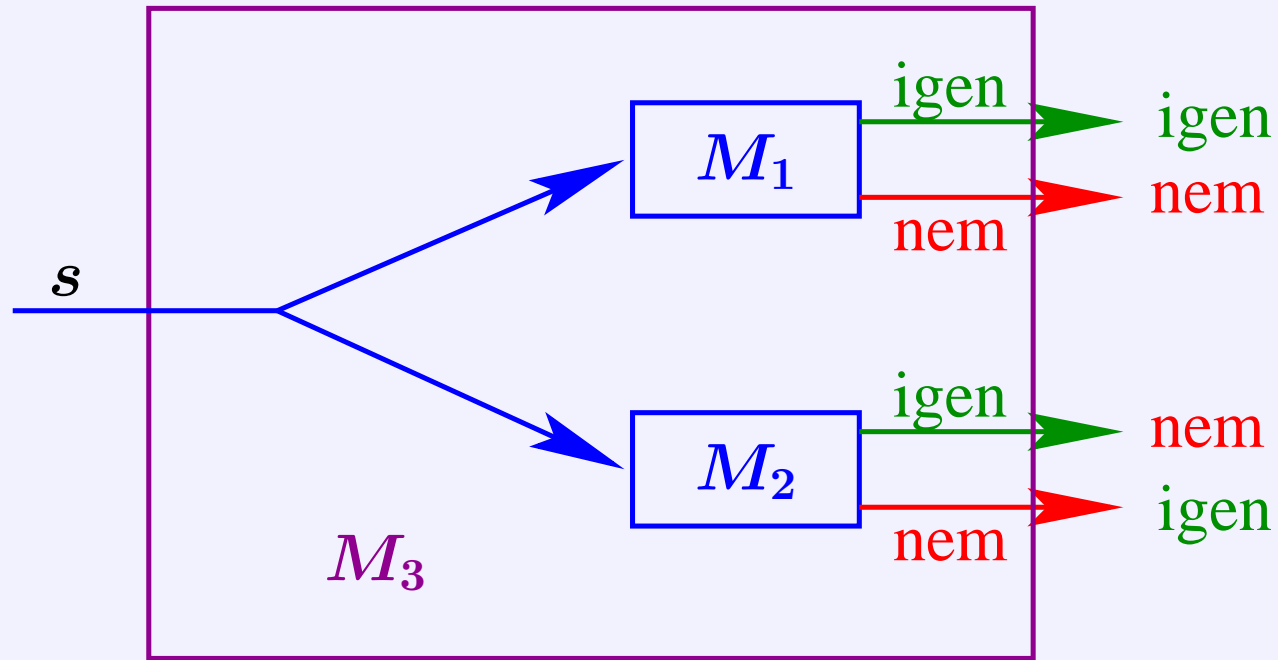
Másik irány:

Tegyük fel, hogy $L \in \mathcal{R}\mathcal{E} \cap \text{co}\mathcal{R}\mathcal{E} \implies$ legyen M_1 , illetve M_2 két TG, melyek az L , illetve az $I^* \setminus L$ nyelvet ismerik fel.

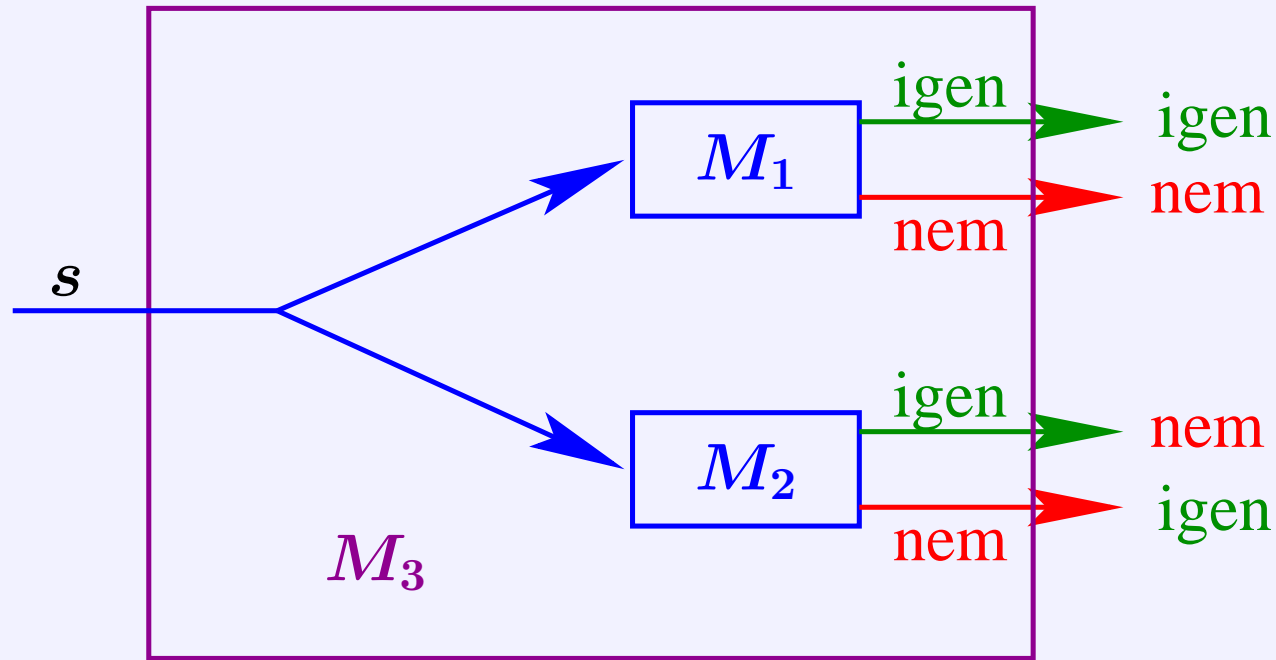
Egy mindig megálló M_3 TG-et szerkesztünk, melyre $L = L_{M_3}$:



Az M_3 pontosan akkor álljon meg, ha M_1 és M_2 valamelyike megáll.



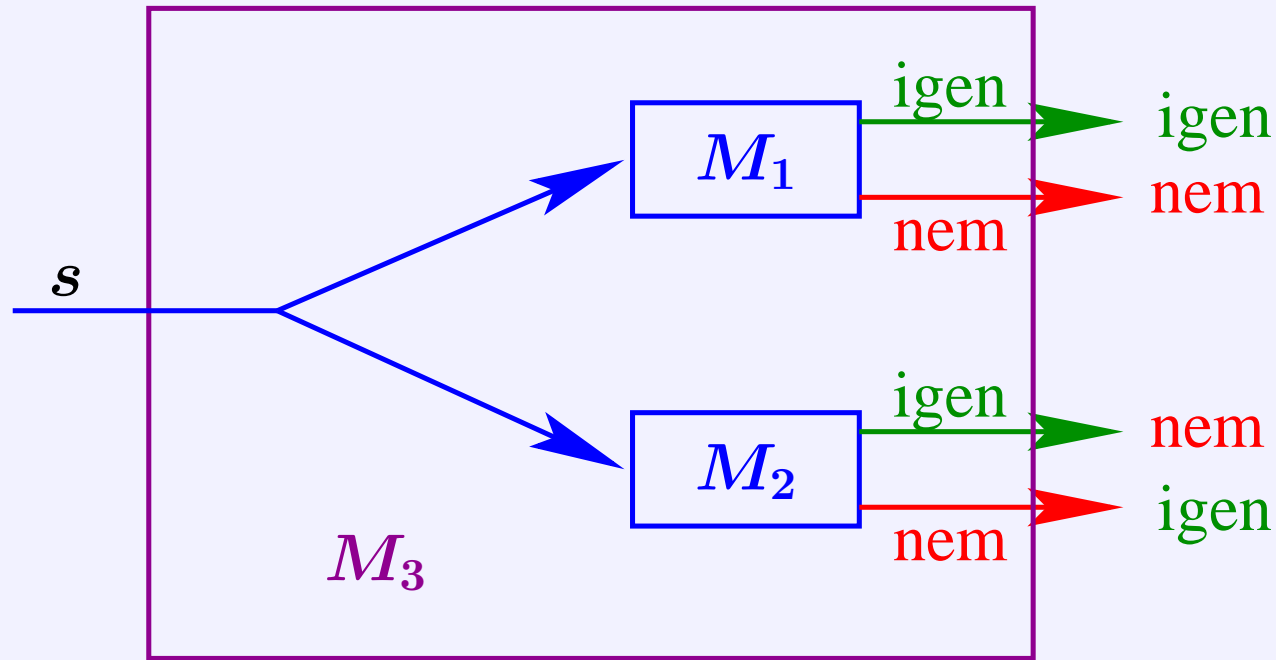
Az M_3 pontosan akkor álljon meg, ha M_1 és M_2 valamelyike megáll. M_3 akkor fogad el, ha a megállás M_1 elfogadó, vagy pedig M_2 elutasító állapotában történt.



Az M_3 pontosan akkor álljon meg, ha M_1 és M_2 valamelyike megáll.

M_3 akkor fogad el, ha a megállás M_1 elfogadó, vagy pedig M_2 elutasító állapotában történt.

$\implies M_3$ az L nyelvet ismeri fel és mindig megáll



Az M_3 pontosan akkor álljon meg, ha M_1 és M_2 valamelyike megáll.
 M_3 akkor fogad el, ha a megállás M_1 elfogadó, vagy pedig M_2 elutasító állapotában történt.

$\implies M_3$ az L nyelvet ismeri fel és mindig megáll

M_3 megvalósítható párhuzamosság nélkül is $\implies M_3$ felváltva lépteti M_1 -et és M_2 -t. ✓

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $\text{Im}(f) = L$).

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $\text{Im}(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $\text{Im}(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $Im(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Kivétel: \implies ha M elutasító állapotban áll meg, akkor M' végtelen ciklusba esik.

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $Im(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Kivétel: \implies ha M elutasító állapotban áll meg, akkor M' végtelen ciklusba esik.

$\implies M'$ gép csak az $s \in L$ szavakra áll meg; megálláskor s lesz az output szalagon. \implies

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $\text{Im}(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Kivétel: \implies ha M elutasító állapotban áll meg, akkor M' végtelen ciklusba esik.

$\implies M'$ gép csak az $s \in L$ szavakra áll meg; megálláskor s lesz az output szalagon. \implies

Az M' által kiszámított $f_{M'}$ függvény értékkészlete L . ✓

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $Im(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Kivétel: \implies ha M elutasító állapotban áll meg, akkor M' végtelen ciklusba esik.

$\implies M'$ gép csak az $s \in L$ szavakra áll meg; megálláskor s lesz az output szalagon. \implies

Az M' által kiszámított $f_{M'}$ függvény értékkészlete L . \checkmark

\Leftarrow : Tegyük fel, hogy f egy parciálisan rekurzív függvény, $f = f_M$, ahol M egy TG.

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $Im(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Kivétel: \implies ha M elutasító állapotban áll meg, akkor M' végtelen ciklusba esik.

$\implies M'$ gép csak az $s \in L$ szavakra áll meg; megálláskor s lesz az output szalagon. \implies

Az M' által kiszámított $f_{M'}$ függvény értékkészlete L . \checkmark

\Leftarrow : Tegyük fel, hogy f egy parciálisan rekurzív függvény, $f = f_M$, ahol M egy TG.

Tekintsük az I^* szavainak $w_0, \dots, w_n \dots$ kanonikus felsorolását.

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $Im(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Kivétel: \implies ha M elutasító állapotban áll meg, akkor M' végtelen ciklusba esik.

$\implies M'$ gép csak az $s \in L$ szavakra áll meg; megálláskor s lesz az output szalagon. \implies

Az M' által kiszámított $f_{M'}$ függvény értékkészlete L . \checkmark

\Leftarrow : Tegyük fel, hogy f egy parciálisan rekurzív függvény, $f = f_M$, ahol M egy TG.

Tekintsük az I^* szavainak $w_0, \dots, w_n \dots$ kanonikus felsorolását.

Ki kellene próbálni minden w_i inputtal, hogy hátha pont s -et a keresett szót számolja ki M .

Függvények és halmazok

Tétel. Az $L \subseteq I^*$ nyelv akkor és csak akkor rekurzíve felsorolható, ha van olyan $f : I^* \rightarrow I^*$ parciálisan rekurzív függvény, melynek értékkészlete éppen az L nyelv (szokásos jelöléssel $Im(f) = L$).

Bizonyítás: \Rightarrow : Tegyük fel, hogy L rekurzíve felsorolható $\implies \exists M$ TG, melyre $L = L_M$.

\implies Legyen M' olyan TG, mely kezdetben az $s \in I^*$ bemenő szót felmásolja az output szalagjára, azután szimulálja az M gépet.

Kivétel: \implies ha M elutasító állapotban áll meg, akkor M' végtelen ciklusba esik.

$\implies M'$ gép csak az $s \in L$ szavakra áll meg; megálláskor s lesz az output szalagon. \implies

Az M' által kiszámított $f_{M'}$ függvény értékkészlete L . ✓

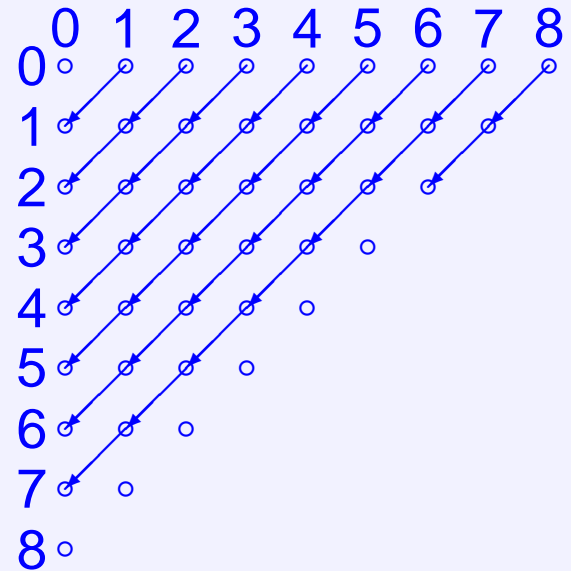
\Leftarrow : Tegyük fel, hogy f egy parciálisan rekurzív függvény, $f = f_M$, ahol M egy TG.

Tekintsük az I^* szavainak $w_0, \dots, w_n \dots$ kanonikus felsorolását.

Ki kellene próbálni minden w_i inputtal, hogy hátha pont s -et a keresett szót számolja ki M .

Baj van, ha valamelyik szóra végtelen ciklusba kerül.

Baj van, ha valamelyik szóra végtelen ciklusba kerül.
A természetes számokból álló párok is felsorolhatók:



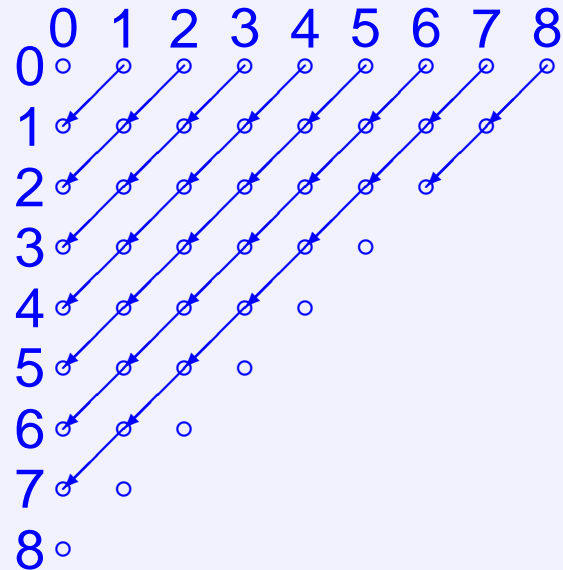
Baj van, ha valamelyik szóra végtelen ciklusba kerül.
A természetes számokból álló párok is felsorolhatók:



Az M' TG az (i, j) párok sorozata szerint megy sorba. Tegyük fel, hogy $s \in I^*$ az M' bemenete.

Baj van, ha valamelyik szóra végtelen ciklusba kerül.

A természetes számokból álló párok is felsorolhatók:

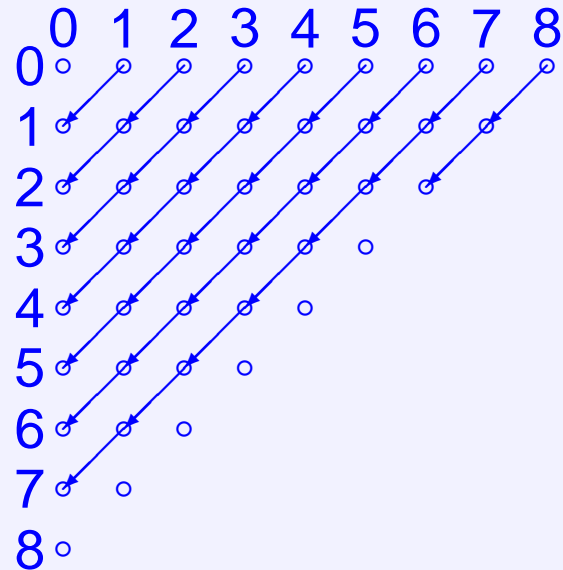


Az M' TG az (i, j) párok sorozata szerint megy sorba. Tegyük fel, hogy $s \in I^*$ az M' bemenete.

$\implies M'$ szimulálja az M első $\leq i$ lépését a w_j szón.

Baj van, ha valamelyik szóra végtelen ciklusba kerül.

A természetes számokból álló párok is felsorolhatók:



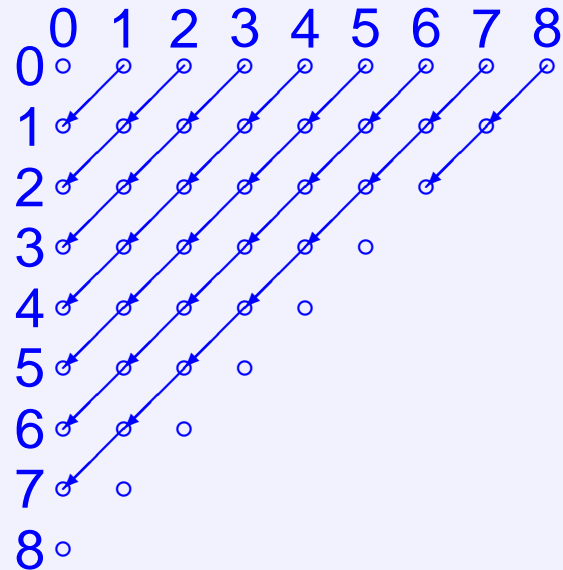
Az M' TG az (i, j) párok sorozata szerint megy sorba. Tegyük fel, hogy $s \in I^*$ az M' bemenete.

$\implies M'$ szimulálja az M első $\leq i$ lépését a w_j szón.

Ha M megáll és o outputot produkál, akkor ellenőrzi, hogy $s = o$ teljesül-e.

Baj van, ha valamelyik szóra végtelen ciklusba kerül.

A természetes számokból álló párok is felsorolhatók:



Az M' TG az (i, j) párok sorozata szerint megy sorba. Tegyük fel, hogy $s \in I^*$ az M' bemenete.

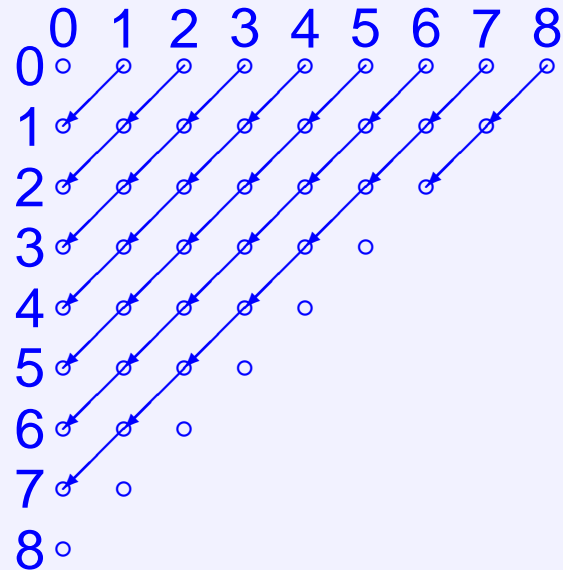
$\implies M'$ szimulálja az M első $\leq i$ lépését a w_j szón.

Ha M megáll és o outputot produkál, akkor ellenőrzi, hogy $s = o$ teljesül-e.

IGEN $\rightarrow M'$ megáll és elfogadja s -et.

Baj van, ha valamelyik szóra végtelen ciklusba kerül.

A természetes számokból álló párok is felsorolhatók:



Az M' TG az (i, j) párok sorozata szerint megy sorba. Tegyük fel, hogy $s \in I^*$ az M' bemenete.

$\implies M'$ szimulálja az M első $\leq i$ lépését a w_j szón.

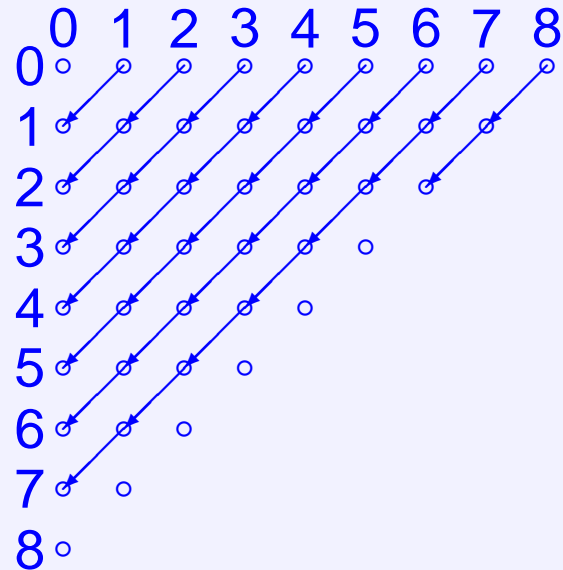
Ha M megáll és o outputot produkál, akkor ellenőrzi, hogy $s = o$ teljesül-e.

IGEN $\rightarrow M'$ megáll és elfogadja s -et.

NEM (nem áll meg i lépésen belül, vagy az eredmény nem s) \implies

Baj van, ha valamelyik szóra végtelen ciklusba kerül.

A természetes számokból álló párok is felsorolhatók:



Az M' TG az (i, j) párok sorozata szerint megy sorba. Tegyük fel, hogy $s \in I^*$ az M' bemenete.

$\implies M'$ szimulálja az M első $\leq i$ lépését a w_j szón.

Ha M megáll és o outputot produkál, akkor ellenőrzi, hogy $s = o$ teljesül-e.

IGEN $\rightarrow M'$ megáll és elfogadja s -et.

NEM (nem áll meg i lépésen belül, vagy az eredmény nem s) \implies következő pár

Mi lesz az M' gép $L_{M'}$ nyelve?

Mi lesz az M' gép $L_{M'}$ nyelve?

$$L_{M'} \subseteq \text{Im}(f) \quad \checkmark$$

Mi lesz az M' gép $L_{M'}$ nyelve?

$$L_{M'} \subseteq \text{Im}(f) \quad \checkmark$$

Ha $s \in \text{Im}(f) \implies \exists j$, hogy $s = f_M(w_j)$

Mi lesz az M' gép $L_{M'}$ nyelve?

$$L_{M'} \subseteq \text{Im}(f) \quad \checkmark$$

Ha $s \in \text{Im}(f) \implies \exists j$, hogy $s = f_M(w_j)$

\implies ha M a w_j bemeneten i lépésben kapja meg az s eredményt $\implies M'$
az (i, j) pár feldolgozásakor elfogadja s -et

Mi lesz az M' gép $L_{M'}$ nyelve?

$$L_{M'} \subseteq \text{Im}(f) \quad \checkmark$$

Ha $s \in \text{Im}(f) \implies \exists j$, hogy $s = f_M(w_j)$

\implies ha M a w_j bemeneten i lépésben kapja meg az s eredményt $\implies M'$
az (i, j) pár feldolgozásakor elfogadja s -et

$$\implies L_{M'} \supseteq \text{Im}(f) \quad \checkmark$$

Mi lesz az M' gép $L_{M'}$ nyelve?

$L_{M'} \subseteq \text{Im}(f)$ ✓

Ha $s \in \text{Im}(f) \implies \exists j$, hogy $s = f_M(w_j)$

\implies ha M a w_j bemeneten i lépésben kapja meg az s eredményt $\implies M'$ az (i, j) pár feldolgozásakor elfogadja s -et

$\implies L_{M'} \supseteq \text{Im}(f)$ ✓

Definíció. Egy $L \subseteq I^*$ nyelv **karakterisztikus függvénye**, χ_L a következő:

$$\chi_L(s) = \begin{cases} 1 \in I^*, & \text{ha } s \in L \\ 0 \in I^*, & \text{ha } s \notin L \end{cases}$$

Tétel. Az $L \subseteq I^*$ nyelv pontosan akkor rekurzív, ha χ_L egy rekurzív függvény.

Bizonyítás: \implies : M' írjon ki a végén 0-t vagy 1-et ✓

Mi lesz az M' gép $L_{M'}$ nyelve?

$L_{M'} \subseteq \text{Im}(f)$ ✓

Ha $s \in \text{Im}(f) \implies \exists j$, hogy $s = f_M(w_j)$

\implies ha M a w_j bemeneten i lépésben kapja meg az s eredményt $\implies M'$ az (i, j) pár feldolgozásakor elfogadja s -et

$\implies L_{M'} \supseteq \text{Im}(f)$ ✓

Definíció. Egy $L \subseteq I^*$ nyelv **karakterisztikus függvénye**, χ_L a következő:

$$\chi_L(s) = \begin{cases} 1 \in I^*, & \text{ha } s \in L \\ 0 \in I^*, & \text{ha } s \notin L \end{cases}$$

Tétel. Az $L \subseteq I^*$ nyelv pontosan akkor rekurzív, ha χ_L egy rekurzív függvény.

Bizonyítás: \implies : M' írjon ki a végén 0-t vagy 1-et ✓

\Leftarrow : Ha 1-et ír ki \rightarrow menjen elfogadóba, ha 0-t \rightarrow elutasítóba ✓

Eldönthetetlen problémák

Definíció. Az $L \subseteq I^*$ nyelvet **eldönthetetlen nyelvnek** nevezzük, ha L nem rekurzív.

Eldönthetetlen problémák

Definíció. Az $L \subseteq I^*$ nyelvet **eldönthetetlen nyelvnek** nevezzük, ha L nem rekurzív.

Definíció. **Megállási probléma:** Megáll-e egy TG egy adott inputon?

$$L_h = \left\{ w\#s \in I^* \mid \begin{array}{l} \text{az } M_w \text{ gép létezik, és az } s \text{ bemenettel} \\ \text{elindítva véges sok lépésben megáll} \end{array} \right\}.$$

Eldönthetetlen problémák

Definíció. Az $L \subseteq I^*$ nyelvet **eldönthetetlen nyelvnek** nevezzük, ha L nem rekurzív.

Definíció. **Megállási probléma:** Megáll-e egy TG egy adott inputon?

$$L_h = \left\{ w\#s \in I^* \mid \begin{array}{l} \text{az } M_w \text{ gép létezik, és az } s \text{ bemenettel} \\ \text{elindítva véges sok lépésben megáll} \end{array} \right\}.$$

Tétel. $L_h \in \mathcal{RE} \setminus \mathcal{R}$.

Eldönthetetlen problémák

Definíció. Az $L \subseteq I^*$ nyelvet **eldönthetetlen nyelvnek** nevezzük, ha L nem rekurzív.

Definíció. **Megállási probléma:** Megáll-e egy TG egy adott inputon?

$$L_h = \left\{ w\#s \in I^* \mid \begin{array}{l} \text{az } M_w \text{ gép létezik, és az } s \text{ bemenettel} \\ \text{elindítva véges sok lépésben megáll} \end{array} \right\}.$$

Tétel. $L_h \in \mathcal{RE} \setminus \mathcal{R}$.

Bizonyítás: $L_h \in \mathcal{RE}$: Vegyünk egy univerzális Turing-gépet, amit kicsit módosítunk: **ha megáll menjen át elfogadóba**

Eldönthetetlen problémák

Definíció. Az $L \subseteq I^*$ nyelvet **eldönthetetlen nyelvnek** nevezzük, ha L nem rekurzív.

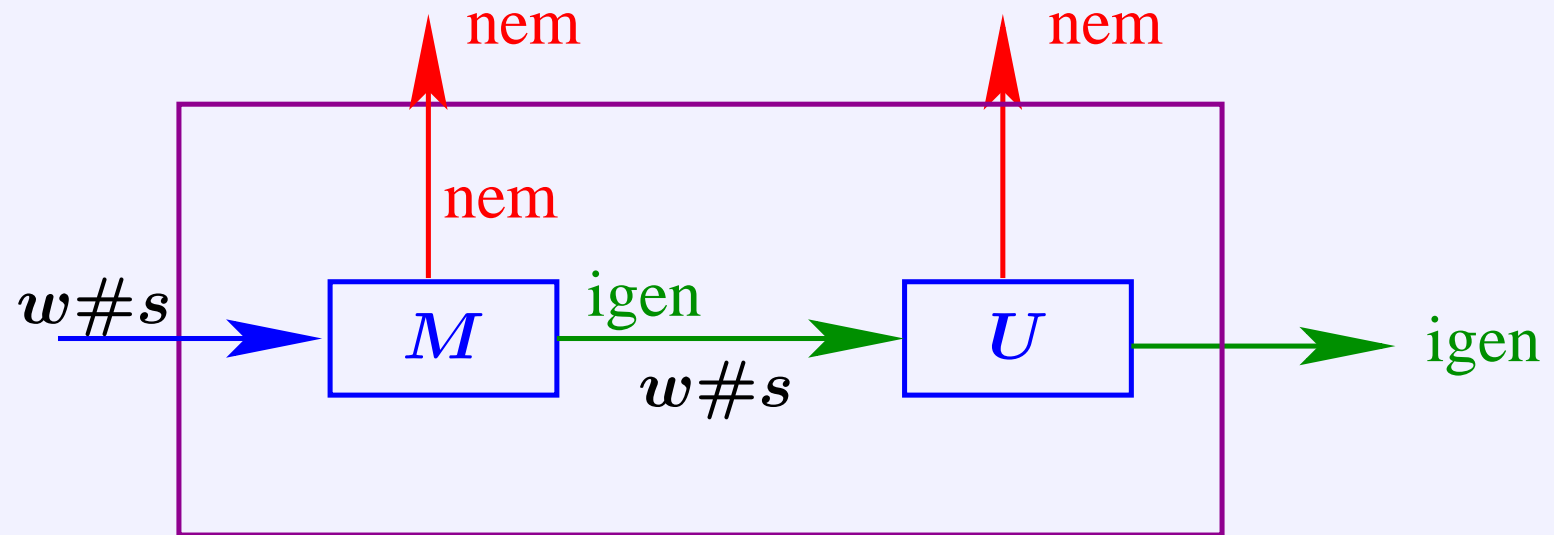
Definíció. **Megállási probléma:** Megáll-e egy TG egy adott inputon?

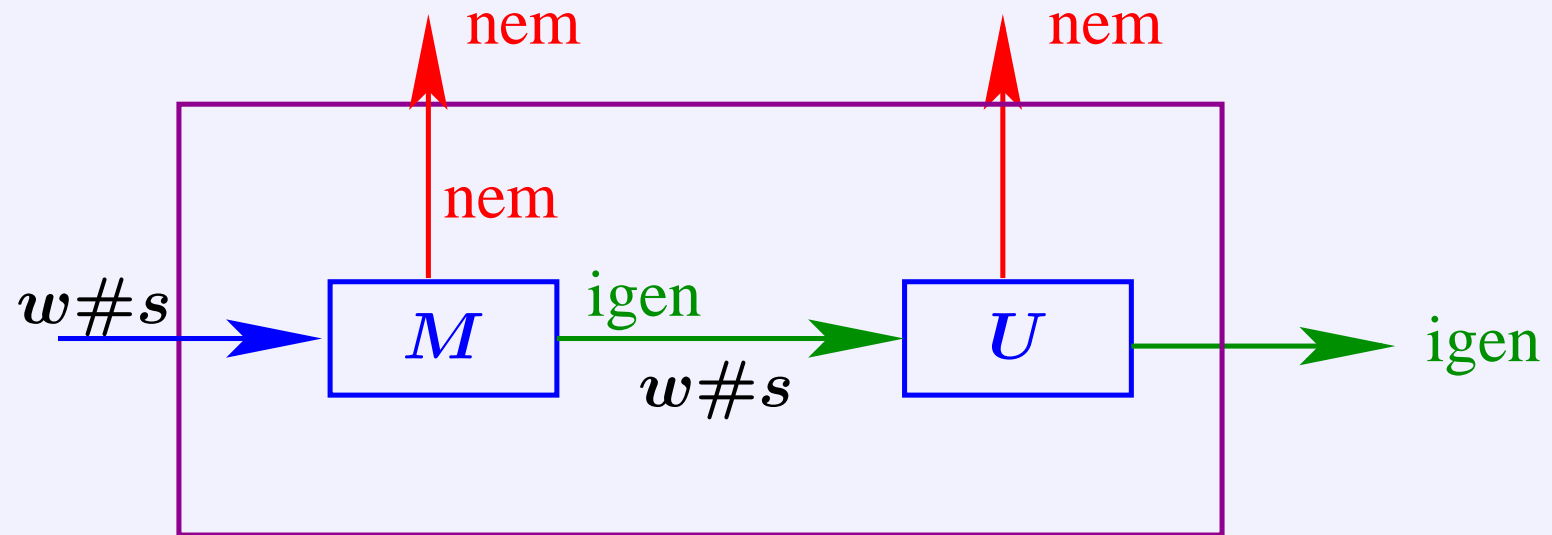
$$L_h = \left\{ w\#s \in I^* \mid \begin{array}{l} \text{az } M_w \text{ gép létezik, és az } s \text{ bemenettel} \\ \text{elindítva véges sok lépésben megáll} \end{array} \right\}.$$

Tétel. $L_h \in \mathcal{RE} \setminus \mathcal{R}$.

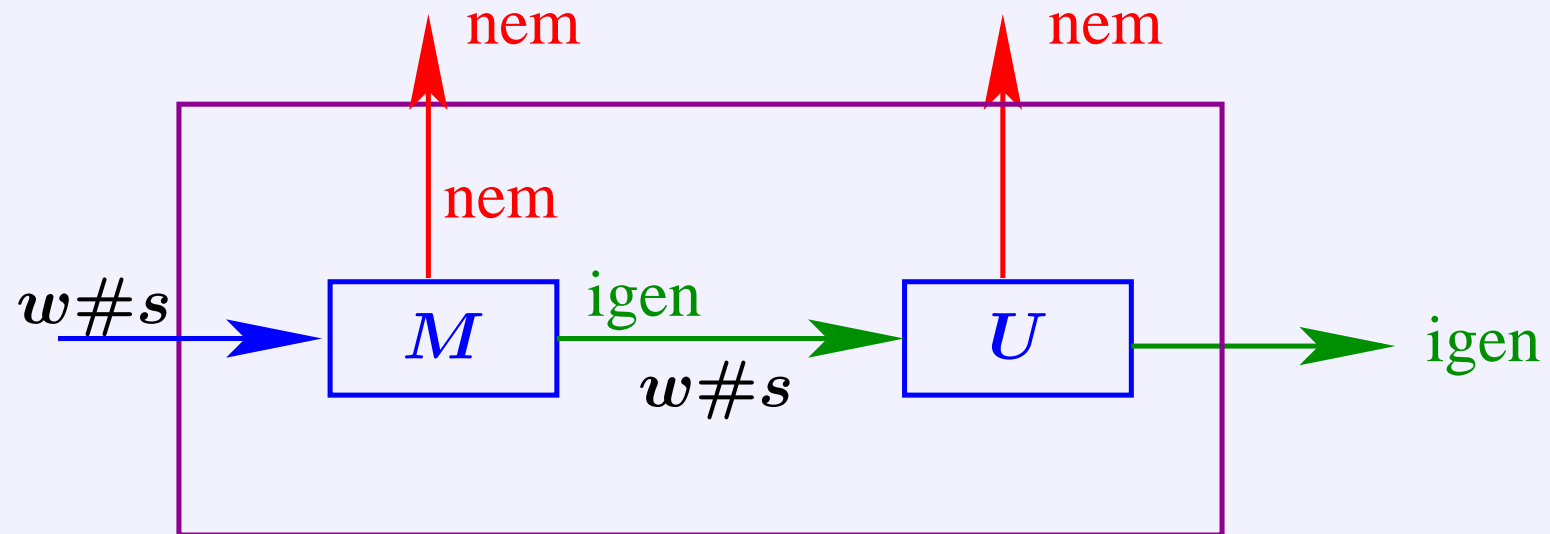
Bizonyítás: $L_h \in \mathcal{RE}$: Vegyünk egy univerzális Turing-gépet, amit kicsit módosítunk: **ha megáll menjen át elfogadóba**

$L_h \notin \mathcal{R}$: Indikekt, tegyük fel, hogy rekurzív $\implies \exists M$ TG, ami felismeri és mindig megáll.





Ez gép L_u -t felismeri és mindig megáll ⚡

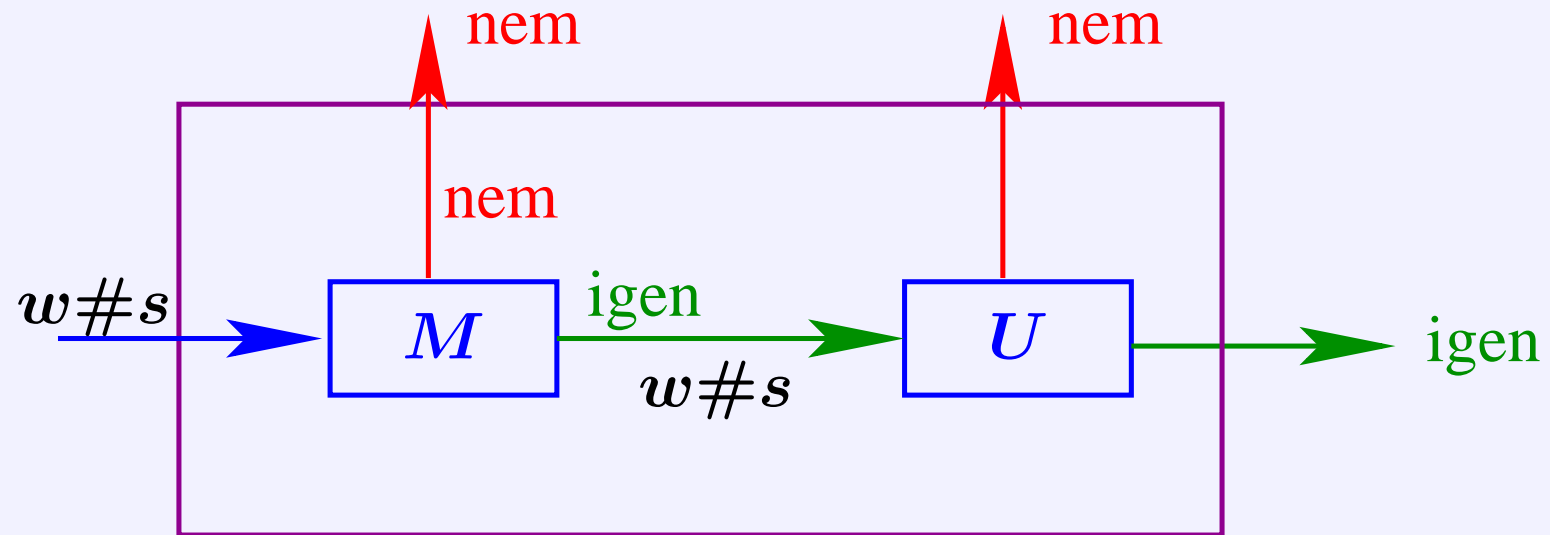


Ez gép L_u -t felismeri és mindig megáll ⚡

Tétel. [A. Church, 1936] Legyen

$$L_\epsilon = \{w \in I^* : M_w \text{ létezik és az } \epsilon \text{ (üres) inputon megáll}\}.$$

$$L_\epsilon \in \mathcal{RE} \setminus \mathcal{R}.$$



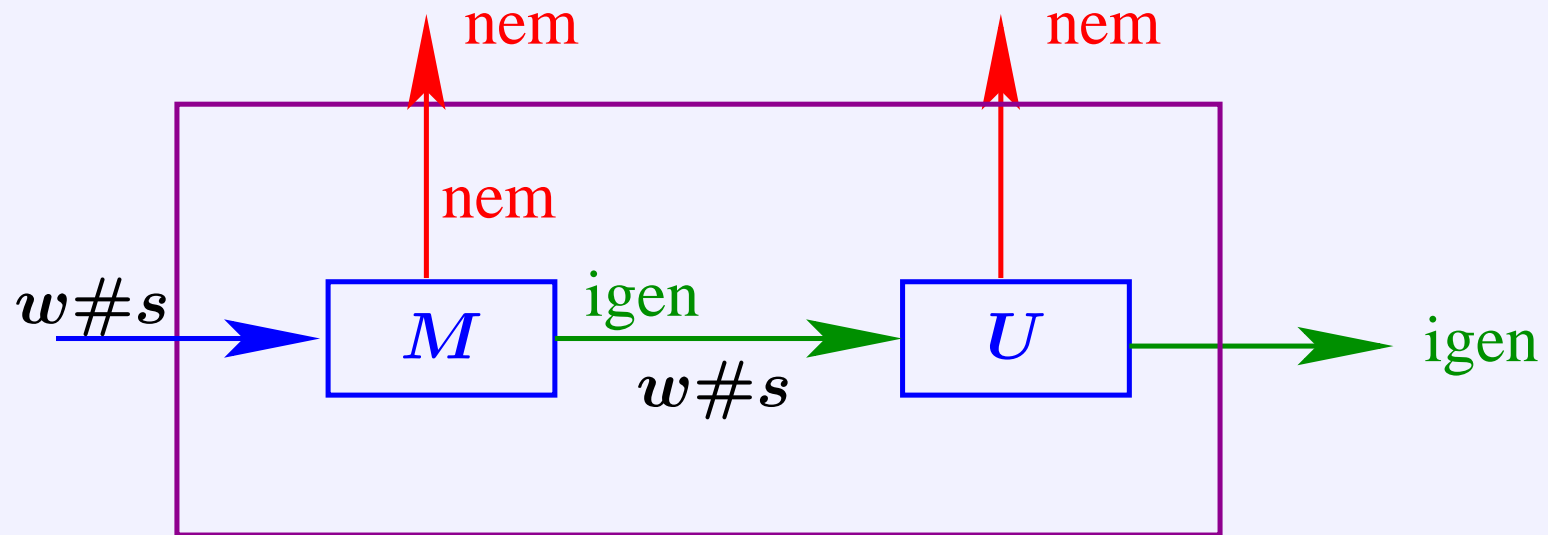
Ez gép L_u -t felismeri és mindig megáll ⚡

Tétel. [A. Church, 1936] Legyen

$$L_\epsilon = \{w \in I^* : M_w \text{ létezik és az } \epsilon \text{ (üres) inputon megáll}\}.$$

$$L_\epsilon \in \mathcal{RE} \setminus \mathcal{R}.$$

Bizonyítás: $L_\epsilon \in \mathcal{RE}$: Az üres inputtal futassuk az L_h -t felismerő gépet ✓



Ez gép L_u -t felismeri és mindig megáll ⚡

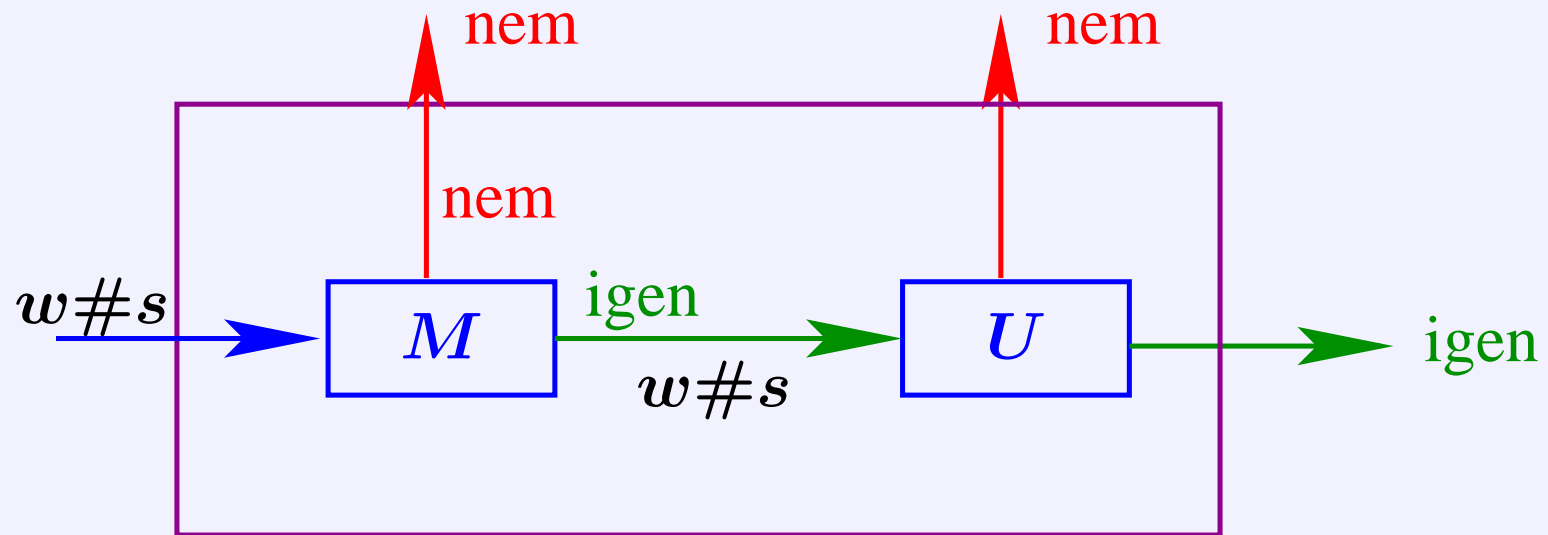
Tétel. [A. Church, 1936] Legyen

$$L_\epsilon = \{w \in I^* : M_w \text{ létezik és az } \epsilon \text{ (üres) inputon megáll}\}.$$

$$L_\epsilon \in \mathcal{RE} \setminus \mathcal{R}.$$

Bizonyítás: $L_\epsilon \in \mathcal{RE}$: Az üres inputtal futassuk az L_h -t felismerő gépet ✓

$L_\epsilon \notin \mathcal{R}$: Belátjuk, hogy ha L_ϵ rekurzív ($\exists M$, ami felismeri és mindig megáll), akkor L_h is.



Ez gép L_u -t felismeri és mindig megáll \downarrow

Tétel. [A. Church, 1936] Legyen

$$L_\epsilon = \{w \in I^* : M_w \text{ létezik és az } \epsilon \text{ (üres) inputon megáll}\}.$$

$$L_\epsilon \in \mathcal{RE} \setminus \mathcal{R}.$$

Bizonyítás: $L_\epsilon \in \mathcal{RE}$: Az üres inputtal futassuk az L_h -t felismerő gépet \checkmark

$L_\epsilon \notin \mathcal{R}$: Belátjuk, hogy ha L_ϵ rekurzív ($\exists M$, ami felismeri és mindig megáll), akkor L_h is.

Ha L_h bemenete $w\#s$, akkor olyan M' -t konstruálunk, aminek belső állapotaiba kódoljuk az s inputot. \checkmark

Hilbert 10. problémája

Legyen $f(x_1, \dots, x_m)$ egész együtthatós m változós polinom:

$$f(x_1, \dots, x_m) = \sum_{i_1=0}^{n_1} \cdots \sum_{i_m=0}^{n_m} a_{i_1 \dots i_m} x_1^{i_1} \cdots x_m^{i_m}$$

Hilbert 10. problémája

Legyen $f(x_1, \dots, x_m)$ egész együtthatós m változós polinom:

$$f(x_1, \dots, x_m) = \sum_{i_1=0}^{n_1} \cdots \sum_{i_m=0}^{n_m} a_{i_1 \dots i_m} x_1^{i_1} \cdots x_m^{i_m}$$

Az f polinom *foka* az előző felírásban előforduló legnagyobb kitevőösszeg:

$$\deg f = \max\{i_1 + \dots + i_m \mid a_{i_1 \dots i_m} \neq 0\}.$$

Hilbert 10. problémája

Legyen $f(x_1, \dots, x_m)$ egész együtthatós m változós polinom:

$$f(x_1, \dots, x_m) = \sum_{i_1=0}^{n_1} \cdots \sum_{i_m=0}^{n_m} a_{i_1 \dots i_m} x_1^{i_1} \cdots x_m^{i_m}$$

Az f polinom *foka* az előző felírásban előforduló legnagyobb kitevőösszeg:

$$\deg f = \max\{i_1 + \dots + i_m \mid a_{i_1 \dots i_m} \neq 0\}.$$

Az

$$(*) \quad f(x_1, \dots, x_m) = 0$$

alakú egyenleteket *diofantikus egyenleteknek* nevezzük.

Hilbert 10. problémája

Legyen $f(x_1, \dots, x_m)$ egész együtthatós m változós polinom:

$$f(x_1, \dots, x_m) = \sum_{i_1=0}^{n_1} \cdots \sum_{i_m=0}^{n_m} a_{i_1 \dots i_m} x_1^{i_1} \cdots x_m^{i_m}$$

Az f polinom *foka* az előző felírásban előforduló legnagyobb kitevőösszeg:

$$\deg f = \max\{i_1 + \dots + i_m \mid a_{i_1 \dots i_m} \neq 0\}.$$

Az

$$(*) \quad f(x_1, \dots, x_m) = 0$$

alakú egyenleteket *diofantikus egyenleteknek* nevezzük.

A (*) diofantikus egyenlet *megoldásán* egy olyan $(u_1, \dots, u_m) \in \mathbb{Z}^m$ egészekből álló m -est értünk, melyre $f(u_1, \dots, u_m) = 0$.

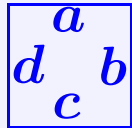
Van-e megoldása egy adott diofantoszi egyenletnek?

Van-e megoldása egy adott diofantoszi egyenletnek?

Tétel. [Matijaszevics, 1970] *Ez eldönthetetlen probléma.*

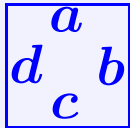
A Dominóprobléma

Dominó:

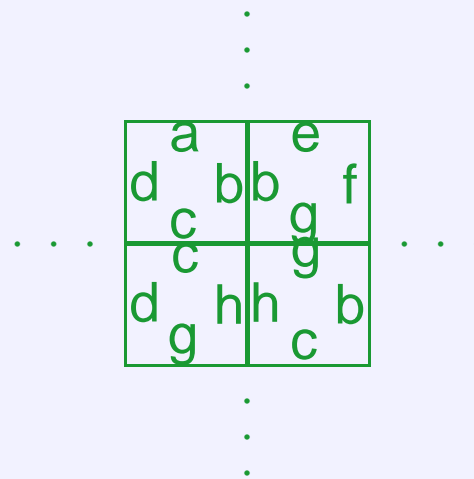


A Dominóprobléma

Dominó:



Egymásmellé rakás:

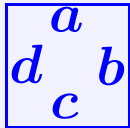


Forgatni nem szabad!

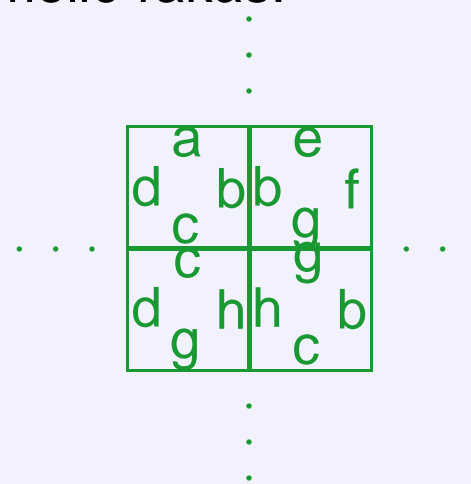
Dominóprobléma: Adott dominó-típusok egy véges \mathcal{F} halmaza; eldöntendő, hogy a sík lefedhető-e hézagtalanul szabályosan illeszkedő \mathcal{F} -beli típusú dominókkal.

A Dominóprobléma

Dominó:



Egymásmellé rakás:



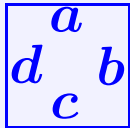
Forgatni nem szabad!

Dominóprobléma: Adott dominó-típusok egy véges \mathcal{F} halmaza; eldöntendő, hogy a sík lefedhető-e hézagtalanul szabályosan illeszkedő \mathcal{F} -beli típusú dominókkal. $\implies D$ nyelv

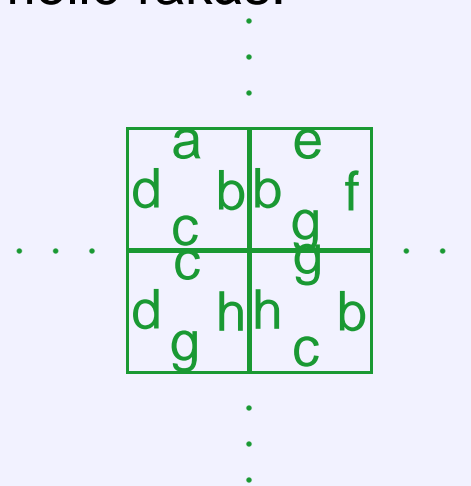
Tétel. $D \notin \mathcal{R}$ és $D \in co\mathcal{RE}$.

A Dominóprobléma

Dominó:



Egymásmellé rakás:



Forgatni nem szabad!

Dominóprobléma: Adott dominó-típusok egy véges \mathcal{F} halmaza; eldöntendő, hogy a sík lefedhető-e hézagtalanul szabályosan illeszkedő \mathcal{F} -beli típusú dominókkal. $\implies D$ nyelv

Tétel. $D \notin \mathcal{R}$ és $D \in co\mathcal{RE}$.

Post megfeleltetési problémája

Emil Post

Post megfeleltetési problémája

Emil Post

Legyen Σ egy véges abc . *Post megfeleltetési problémájának* egy **bemenete** egy (s, t) ($s, t \in \Sigma^*$) alakú rendezett párokból álló véges \mathcal{P} halmaz.

Post megfeleltetési problémája

Emil Post

Legyen Σ egy véges abc . *Post megfeleltetési problémájának* egy **bemenete** egy (s, t) ($s, t \in \Sigma^*$) alakú rendezett párokból álló véges \mathcal{P} halmaz.

A megfeleltetési feladat \mathcal{P} bemenetét *megoldhatónak* nevezzük, ha vannak olyan (nem feltétlenül különböző) \mathcal{P} -beli $(s_1, t_1), (s_2, t_2), \dots, (s_n, t_n)$ párok úgy, hogy

$$s_1 s_2 \cdots s_n = t_1 t_2 \cdots t_n.$$

Post megfeleltetési problémája

Emil Post

Legyen Σ egy véges *abc*. *Post megfeleltetési problémájának* egy **bemenete** egy (s, t) ($s, t \in \Sigma^*$) alakú rendezett párokból álló véges \mathcal{P} halmaz.

A megfeleltetési feladat \mathcal{P} bemenetét *megoldhatónak* nevezünk, ha vannak olyan (nem feltétlenül különböző) \mathcal{P} -beli $(s_1, t_1), (s_2, t_2), \dots, (s_n, t_n)$ párok úgy, hogy

$$s_1 s_2 \cdots s_n = t_1 t_2 \cdots t_n.$$

Ilyenkor az $s_1 s_2 \cdots s_n$, vagy ami ugyanaz, a $t_1 t_2 \cdots t_n$ szót a \mathcal{P} *megoldásának* nevezük.

Post megfeleltetési problémája

Emil Post

Legyen Σ egy véges abc . *Post megfeleltetési problémájának* egy **bemenete** egy (s, t) ($s, t \in \Sigma^*$) alakú rendezett párokból álló véges \mathcal{P} halmaz.

A megfeleltetési feladat \mathcal{P} bemenetét *megoldhatónak* nevezünk, ha vannak olyan (nem feltétlenül különböző) \mathcal{P} -beli $(s_1, t_1), (s_2, t_2), \dots, (s_n, t_n)$ párok úgy, hogy

$$s_1 s_2 \cdots s_n = t_1 t_2 \cdots t_n.$$

Ilyenkor az $s_1 s_2 \cdots s_n$, vagy ami ugyanaz, a $t_1 t_2 \cdots t_n$ szót a \mathcal{P} *megoldásának* nevezük.

Például a $\mathcal{P} = \{(iz, riz), (kar, ka), (ma, ma)\}$ rendszer megoldható. Egy lehetséges megoldás a *karizma* szó.

Tétel. *Ez a probléma eldönthetetlen.*