Linguaggio dell'arresto stretto

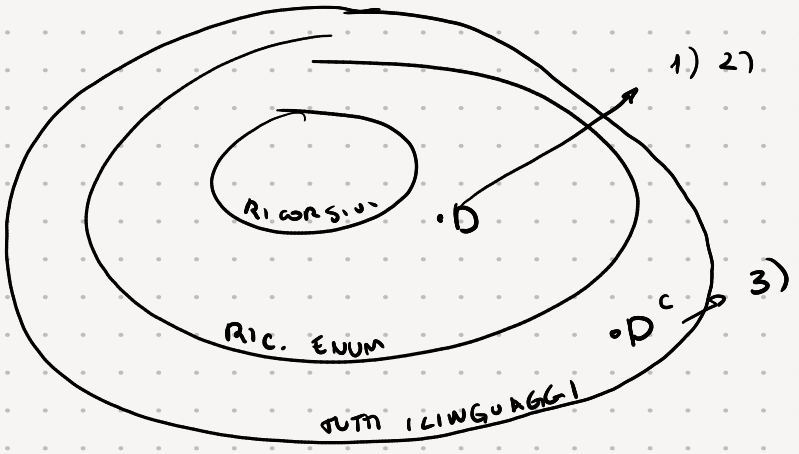
$$D = \{ x \in \{0,1\}^* \mid F_u(x\$x)\downarrow \}$$

Teorema: 1) D è ricorsivamente enum

2) D non è ricorsivo

$$D^c = \{ x \in \{0,1\}^* \mid F_u(x\$x)\uparrow \}$$

3) D^c non è ric. enum.



Punto 1) Si mostra la procedura

Procedura RICNUM ($x \in \{0,1\}^*$)

{

$y = F_u(x\$x)$ → o termina / → non termina

return (1)

}

$F_u(x\$x)\downarrow \Rightarrow x \in D$ e RICNUM$(x) = 1$

$F_u(x\$x)\uparrow \Rightarrow x \notin D$ e RICNUM$(x)\uparrow$

Punto 2) Si dimostra per assurdo

se D è ricorsivo esiste il seguente programma

Procedura ASSURDO A ($x \in \{0,1\}^*$)

e
```
{
  if x ∈ D then return ( 1 - Fu(x $ x)
           else return (0)
}
```

Valutando ASSURDOA(e) otteniamo delle contraddizioni.

È l'esistenza del codice e, parola binaria che codifica ASSURDO A, che ci porta ad ottenere cose impossibili:

-> da un lato per def. di "u"

ASSURDO A(e) = Fu (e $ e)

-> dall'altro lato x def. del programma ASSURDOA to:

$e \in D$ ASSURDO A (e) = 1 - Fu (e $ e)

$e \notin D$ ASSURDO A (e) = 0

⊗ $F_u(e\$e)\downarrow \begin{cases} 0 \\ 1 \end{cases}$ $F_u(e\$e) = 1 - F_u(e\$e)$

$0 = 1$ o $1 = 0$

• $F_u(e\$e)\uparrow$ $\uparrow F_u(e\$e) = 0$

$\uparrow = 0$

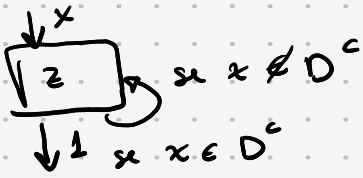
Punto 3) $D^c = \{ x \in \{0,1\}^* \mid F_u(x\$x)\uparrow \}$

non è ricorsivamente enumerabile

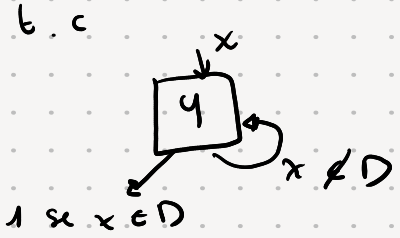
dimostrazione: per assurdo

Suppongo che D^c sia ric. enum.

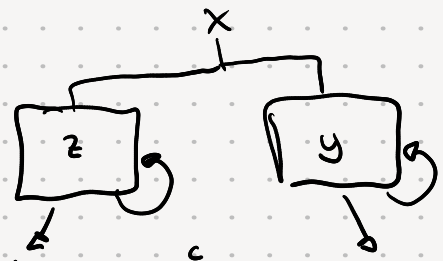
$\Rightarrow$ esiste una procedura z t.c.



Inoltre so che D è ricorsivamente enum (punto 1) e pertanto esiste una proc. y t.c



Idea:



PROGRAMMA ASSURDO B ($x \in \{0,1\}^*$)

{

$k = 1$

while ($z(x)$ non termina in k passi $\wedge$ (and) $y(x)$ non termina in k passi)

{ $k = k + 1$ }

if (ha terminato y) then return (1)

else return (0)

}

Domanda: Cosa fa ASSURDOB(x)?

$x \in D$ $\Rightarrow$ Esiste un k t.c. $y(x)$ termina

$\Rightarrow$ return(1) $\Rightarrow$ ASSURDOB(x) = 1

$x \notin D$ $\Rightarrow$ Esiste un k t.c. $z(x)$ termina

$\Rightarrow$ return(0) $\Rightarrow$ ASSURDOB(x) = 0

$\Downarrow$

ASSURDOB(x) è un algoritmo per D

per D

$\Downarrow$

D è un linguaggio ricorsivo

↑ Contraddizione punto 2)

Z non esiste e D^c non è ricorsivamente enumerabile

Domanda: Esiste un problema che non ammette soluzione algoritmica?

$\Downarrow$

Esiste un problema indecidibile?

$\Downarrow$

Esiste un linguaggio non ricorsivo?

Risposta: Sì

Problema dell'arresto (o fermata)

Input: $w\$x$

Output: Risposta alla domanda $F_w(x)\downarrow$?

$$A = \{ w\$x \in \{0,1\}^* \mid F_u(x)\downarrow \}$$

LINGUAGGIO DELL'ARRESTO

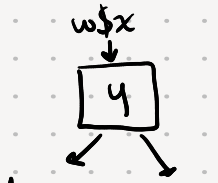
$$D = \{ x \in \{0,1\}^* \mid F_u(x\$x)\downarrow \}$$

$$A = \{ w\$x \in \{0,1\} \mid F_u(w\$x)\downarrow \}$$

Teorema: A non è RICORSIVO

dim: per assurdo

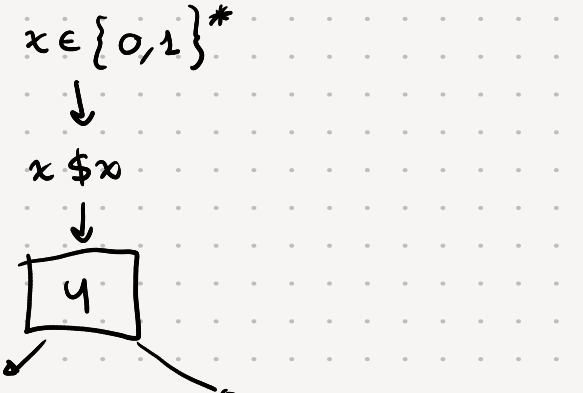
suppongo A ricorsivo e pertanto $\exists$ y t.c.



se $F_u(w\$x)\downarrow$ se $F_u(w\$x)\uparrow$

Allora posso costruire un programma

ASSURDO C (x)



1 se $x \in P$ 0 se $x \notin D$

PROGRAMMA ASSURDOC ($x \in \{0,1\}^*$)

{

IF $x \$ x \in A$ then return 1

else return 0

}

sto supponendo A ricorsivo

Cosa fa ASSURDOC(x)?

$x \in D \Rightarrow \Phi_U(x \$ x) \downarrow \Rightarrow x \$ x \in A$

$\Rightarrow$ ASSURDOC(x) = 1

$x \notin D \Rightarrow \Phi_U(x \$ x) \uparrow \Rightarrow x \$ x \notin A$

$\Rightarrow$ ASSURDOC(x) = 0

⇓

ASSURDOC è un algoritmo per D

⇓

contraddizione perché D è non ricorsivo

⇓

A non è ricorsivo

⇓

Il problema dell'arresto è indecidibile