23/11/2017

BERARDUCCI
TEORIA

Def Una formula si dice in forma normale prenessa se è del tipo

$$\forall \vec{x}\, \exists \vec{y}\, \forall \vec{z}\, \exists \vec{w} \cdots \Theta$$

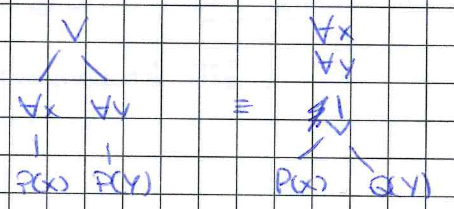
, dove Θ è una formula senza quantificatori

Esempio $\models \forall x\, P(x) \vee \forall y\, Q(y)$ non è in forma normale prenessa, in quanto comincia con una '$\vee$'.
Però questo è equivalente a $\forall x \forall y\, [P(x) \vee Q(y)]$ (lo si vede facilmente a livello semantico), che è in forma normale prenessa.

Esempio $\models \exists x\, P(x) \vee \exists x\, Q(x) \equiv \exists x\, [P(x) \vee Q(x)]$

$\models \exists x\, P(x) \wedge \exists x\, Q(x) \equiv \exists x \exists y\, [P(x) \wedge Q(y)]$

oss La cosa diventa ancora più evidente se si considerano gli alberi sintattici di quelle formule



Queste regole, unite alle leggi di De Morgan generalizzate sono sufficienti a portare una formula in FNP?

Si considerino anche le seguenti:

· $\models \Theta(\bar{y}) \underset{\vee}{\wedge} \underset{\exists}{\forall} x\, P(x,\bar{y}) \equiv \underset{\exists}{\forall} x\, [\Theta(\bar{y}) \underset{\vee}{\wedge} P(x,\bar{y})]$

oss Si analizzi invece il caso in cui Θ dipende da x.

$\models \Theta(\bar{x},\bar{y}) \underset{\vee}{\wedge} \underset{\exists}{\forall} x\, P(\bar{x},\bar{y})$

In realtà la seconda x 'è muta' quindi potrei anche chiamarla z
In questo modo ci si riduce al caso precedente, ottenendo

$\models \Theta(\bar{x},\bar{y}) \underset{\vee}{\wedge} \underset{\exists}{\forall} z\, P(z,\bar{y})$

oss Una volta ottenuta la FNP, volendo, sarebbe possibile mettere la formula senza quantificatori in CNF o DNF.

esempio $(\exists x\, P(x)) \longrightarrow Q \equiv \cancel{\forall x}\, \exists x\, (P(x) \to Q)$

$$\neg \exists x\, P(x) \vee Q \equiv (\forall x\, \neg P(x)) \vee Q \equiv \forall x\, (\neg P(x) \vee Q)$$

oss Se fossi nella logica del II ordine, e avessi, ad esempio, $\forall x \exists y\, P(x,y)$

$$\forall x \exists y\, P(x,y) \equiv \exists f\, \forall x\, P(x, f(x))$$

che è un possibile modo per esprimere l'assioma della scelta.

Nel caso di più quantificatori, invece, si potrebbe avere, ad esempio, $\forall x \exists y \forall z \exists w\, P(x,y,z,w)$

$$\forall x \exists y \forall z \exists w\, P(x,y,z,w) \equiv \exists f\, \forall x\, \forall z\, \exists w\, P(x, f(x), z, w) \equiv$$

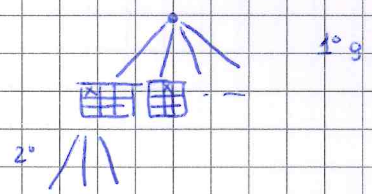
$$\equiv \exists f\, \exists h\, \forall x\, \forall z\, P(x, f(x), z, h(x,w))$$

In generale, indipendente dal numero di quantificatori, le formule si riducono sempre ad avere un certo numero di quantificatori esistenziali su delle funzioni e un certo numero di quantificatori universali sulle variabili.

oss Questo fatto ha un'interpretazione in teoria dei giochi.
Ad esempio, considerando quelle formule nel modo
"per ogni mossa del primo giocatore esiste una contromossa del secondo tale che per ogni mossa del primo ... un modo ~~per~~ da vincere"
In questo modo si otterrebbe una strategia vincente del secondo (se la formula fosse vera).

(In un gioco una strategia di un giocatore è una funzione che mossa fare in base alle mosse dell'avversario (e quindi anche delle mie)).

È possibile vedere il gioco come un albero



Una strategia vincente è una strategia per cui se uno gioca a caso e l'altro la segue, quest'ultimo sicuramente vince.

oss. È vero che c'è una strategia vincente per l'uno o l'altro?

Si consideri $\forall x \exists y \forall z \exists w\ P$

è vera $\Leftrightarrow$ c'è una strategia $\overset{\text{per } \exists}{(f,g)}$, con $y = f(x)$, $w = g(x,z)$

Se non c'è strategia per $\exists$, allora $\neg \forall x \exists y \forall z \exists w\ P \equiv$

$$\equiv \exists x\ \neg \exists y \forall z \exists w\ P \equiv \dots \equiv \exists \underset{c}{x}\ \forall y\ \exists \underset{h(y)}{z}\ \forall w\ \neg P$$

Si ha che (c,h) sono la strategie per $\forall$

(chiaramente si sta ipotizzando che il gioco ammetta sempre un vincitore, cioè non si può pareggiare)

Si ricordi che $\Pi_1^0 = \forall RIC$, $\Sigma_1^0 = \exists RIC$, ..., $\Sigma_{n+1}^0 = \exists \Pi_n^0$, $\Pi_{n+1}^0 = \forall \Sigma_n^0$

~~Si vedano~~

Esercizio
- I quantificatori limitati non cambiano la classe di complessità
- $\exists\exists \equiv \exists$ e $\forall\forall = \forall$
- I connettivi $\wedge$ e $\vee$ non cambiano la complessità
- $\neg\Sigma_n^0 = \Pi_n^0$ e $\neg\Pi_n^0 = \Sigma_n^0$

a) Si supponga di avere $\forall x \leq y\ \exists z\ P(x,y,z)$.

27/11/2017

BERARDUCCI
TEORIA

Si facciamo alcune precisazioni.

Con Σ_n^0 si possono indicare più cose.

Ad esempio, si possono intendere i predicati ($\subseteq \mathbb{N}^k$).

Primitivi ricorsivi $\subseteq$ Ricorsivo $\subseteq \Sigma_1^0 = \exists$ Ric

$\Pi_1^0 = \forall$ Ric, $\Sigma_2^0 = \exists \Pi_1^0$ ...

Inoltre si può parlare anche di Σ_n^0-formule ($\mathcal{L} = \{0, S, +, \cdot\}$)

$$\Delta_0 = t_1 = t_2 \mid \neg \Delta_0 \mid \Delta_0 \hat{\vee} \Delta_0 \mid \forall x \leq t \, \Delta_0 \mid \exists x \leq t \, \Delta_0 \mid$$

$\cap \mid$

Σ_1^0 ... Σ_2^0

Π_1^0 ... Π_2^0 ...

Chiaramente si intende che un predicato è Σ_n^0 se è esprimibile con una Σ_n^0-formula.

Teorema Se f è calcolabile allora esiste $\Theta(\vec{x}, y)$ in $\mathcal{L} = \{0, S, +, \cdot\}$ una Σ_1^0-formula tale che $f(\vec{a}) = b \Leftrightarrow \mathbb{N} \models \Theta(\vec{a}, b)$

Dim

Si dimostra per induzione sulla complessità della calcolabilità di f.

$P(x)$ ricorsivo, $P \subseteq \mathbb{N}$
Esiste $\Theta(x) \in \Sigma_1^0$-formula

$$\forall a \in \mathbb{N} \, [\underbrace{P(a) \Leftrightarrow \mathbb{N} \models \Theta(a)}]$$

χ_P calcolabile $\varphi_P(\vec{x}, y)$ Σ_1^0-formula

$$\chi_P(a) = \underset{\{0,1\}}{b} \Leftrightarrow \mathbb{N} \models \varphi_P(a, b)$$

Si definisce $\Theta(a) = \varphi_P(a, 1)$

Quindi $P(a) \Leftrightarrow \chi_P(a, 1) \Leftrightarrow \mathbb{N} \models \varphi_P(a, 1) \Leftrightarrow \mathbb{N} \models \Theta(a)$

Teorema P predicato Σ_1^0 (cioè semidecidibile, cioè ric. enum), $P = \Gamma(f)$
$f(x) = y \iff P(x,y) \implies f$ è calcolabile.

Dim

- Se P ricorsivo

Si definisce $f(x) = \mu y\, P(x,y) = \mu y\, (h(x,y) = 1)$ $h = \chi_P$

Prop f calcolabile $\iff \Gamma(f)$ è $\exists$ Ric $= \Sigma_1^0$

Dim

"$\Rightarrow$" È la dim. fatta con la β di Gödel e induz. sulla def di calcolabile

"$\Leftarrow$" Idea: $P = \Gamma(f) = \exists$ Ric

Supponiamo che $\Gamma(f) = \{(x,y) \mid \exists t\, Q(x,y,t)\}$

$h(x) = \mu \langle y,t \rangle\, Q(x,y,t)$

$f(x) = \Pi_1(h(x))$ $\Pi_1(y,t) = y$

Oss - $f(\bar{x}, y) = \prod_{z<y} h(x,z)$ h prim ric tot $\Rightarrow f$ lo è

Infatti
$f(\bar{x}, 0) = 1$ è una ric primit.
$f(\bar{x}, y+1) = f(\bar{x}, y) \cdot h(x,y)$

- Quantif limitati

$P(x,y) = \forall z < y\, R(x,y)$

$\chi_P(x,y) = \prod_{z<y} \chi_R(x,y)$ che è calcolabile

- $\neg P(x)$

$\chi_{\bar{P}}(x) = 1 - \chi_P(x)$

Prop $f(\bar{x}, y) = \begin{cases} \mu z < y & P(x,y) \text{ se } \exists z < y\ P(x,y) \\ y & \text{altrimenti} \end{cases}$

P ric prim calc tot $\Rightarrow f$ lo è

Dim

$f(\bar{x}, 0) = 0$

$f(\bar{x}, y+1) = \begin{cases} f(x,y) & \text{se } \exists z<y\ P(x,z) \\ y & \text{se } (\neg \exists z<y\ P(x,z)) \wedge P(x,y) \\ y+1 & \text{altrimenti} \end{cases}$

Le condizioni sono calcol. $\Rightarrow f$ è calc.

Esercizio la succ. di fibonacci è prim ric.

Corollario $f(\bar{x}, y) = \mu z < h(y, x) \; P(x)$

h, P calc tot $\vee$ prim ric $\Rightarrow f(\cdot)$ è

Dim

si considera $\hat{f}(x, y, t) = \mu z < t \; P(x, y)$

$f(x, y) = \hat{f}(x, y, h(x, y))$

Esercizio $i \mapsto p(i)$ i-esimo primo è prim ric.

$p(0) = 2$

$p(n+1) = \mu y (\overbrace{y \text{ primo}}^{\Delta_0} \wedge y > p(n))$

se voglio la calcolabilità è basta ho finito.
se voglio prim ric devo limitare il minimo

si sa che tra n e $n!+1$ c'è un primo
quindi posso limitare

CODIFICHE

Si vuole codificare una successione di naturali con un naturale.

- Una codifica surgettiva sfrutta il fatto che per ogni $(a_0, \dots, a_n)$ $\exists\, c, d$ tale che $\beta(c,d,0) = a_0, \dots, \beta(c,d,n) = a_n$
  Quindi (c,d) codifica $(a_0, \dots, a_n)$
  Ma $(c,d) \leftrightarrow 2^c(2d+1)-1$ quindi $\hat{\beta}(n,i) = \beta(\pi_1(n), \pi_2(n), i)$
- Una codifica iniettiva invece può essere $(a_0, \dots, a_n) = \prod_{i=0}^{n} p_i^{a_i+1}$
  Questa però non è surgettiva ($2^k \cdot 5^0$ non codifica nulla)

  Infatti questa codifica è prim. ric.

  $s \in \mathbb{N}$ codifica una succ. $\Leftrightarrow P(s)$, con P r.c. $\Leftrightarrow$
  $\Leftrightarrow \forall i \leq s \;\; p(i) \mid s \Rightarrow \forall j < i \;\; p(j) \mid s$

Esercizio Codifica biunivoca

0/11/2017

BERARDUCCI
TEORIA

Si ricordi la codifica visto l'oltra volta

$$(a_1, \dots, a_n) = \prod_{i \le n} p_i^{a_i+1} = s$$

OSS $s \in \mathbb{N}$ codifica una successione $\Leftrightarrow \forall i \le s \ \forall j < i \ \ p(i) \mid s \to p(j) \mid s$
Quindi questo è un predicato primitivo ricorsivo

OSS $\mathbb{N}^2 \to \mathbb{N}$ t.c. $(s, i) \mapsto (s)_i \in \mathbb{N}$
Quindi se $s = (a_1, \dots, a_n) \Rightarrow (s)_i = a_i$
Perciò $(s)_i = \begin{cases} \mu u < s \ \ p(i)^{u+1} \mid s \wedge p(i)^{u+2} \nmid s \\ s \end{cases}$ e questa è primitiva ricorsiva
se questo u esiste

OSS Si ragioni ora sulla lunghezza di s.
$\ell(s) = \mu i < s \ \ (s)_{i+1} = s$. Con questa definizione si ha, come si voleva che $\ell((a_0, \dots, a_n)) = n$.

OSS Si può definire $A \frown B = C$ in modo che $(a_0, \dots, a_n) \frown (a_{n+1}, \dots, a_{n+k}) = (a_0, \dots, a_{n+k})$
Quindi

$$A \frown B = C \Leftrightarrow \begin{cases} \forall i < \ell(A) \quad (C)_i = (A_i) \\ \wedge \ \forall i < \ell(B) \quad (C)_{i+\ell(A)} = (B)_i \\ \wedge \ A, B, C \text{ codificano successioni} \\ \\ 0 \quad \text{se } A \vee B \vee C \text{ non codificano} \end{cases}$$

Teorema Sia $f(x) = h(x, (f(0), \dots, f(x-1)))$ se h è μ-ricorsiva, allora f lo è.

Dim

Sia $f^{\#}(x) = (f(0), \dots, f(x-1))$.
Quindi $f(x) = h(x, f^{\#}(x))$.
Si noti che $f^{\#}(0) = ()$

$$f^{\#}(n+1) = (f(0), \dots, f(n)) = (f(0), \dots, f(n-1)) \frown (f(n)) = f^{\#}(n) \frown (f(n))$$
$$= f^{\#}(n) \frown (h(n, f^{\#}(n))) = H(n, f^{\#}(n))$$

dunque $f^{\#}(x)$ è primitiva ricorsiva

CODIFICA DEI TERMINI

Si consideri $\mathcal{L}$ un linguaggio finito, ad esempio $\mathcal{L} = \{0, S, +, \cdot\}$
e si consideri la funzione $\# : L \cup \{\neg, \wedge, \vee, \rightarrow, \forall, \exists, =\} \cup \{v\} \longrightarrow \mathbb{N}$
iniettiva

Si definisca ora $\ulcorner\ \urcorner : L\text{-termini} \rightarrow \mathbb{N}$ iniettiva in modo che

- $\ulcorner f(t_1, \dots, t_n) \urcorner = (\#(f), \ulcorner t_1 \urcorner, \dots, \ulcorner t_n \urcorner)$
- $\ulcorner c \urcorner = (\#(c))$
- $\ulcorner x_i \urcorner = (\#(v), i)$

$\underset{\text{variabile}}{x_i}$

Esempio $\quad \mathcal{L} = \{0, S, +, \cdot\}$

$$\ulcorner 0 \urcorner = (\#(0)) = 2^{\#(0)+1}$$

$$\ulcorner S(0) \urcorner = (\#(S), \ulcorner 0 \urcorner) = 2^{\#(S)+1} \cdot 3^{\ulcorner 0 \urcorner}$$

OSS $\cdot$ La funzione $n \longmapsto \ulcorner S^n(0) \urcorner = num(n)$ è primitiva ricorsiva.

Infatti: $\cdot\ num(0) = \ulcorner 0 \urcorner$
$\cdot\ num(n+1) = \ulcorner S^{n+1}(0) \urcorner = \ulcorner S(S^n(0)) \urcorner = (\#(S), \ulcorner S^n(0) \urcorner) = (\#(S), num(n))$

Teorema $\quad \{\ulcorner t \urcorner \mid t \text{ è un } \mathcal{L}\text{-termine}\}$ è primitivo ricorsivo.

Dim

Sia $T(n) = \begin{cases} 1 & \text{se } n \text{ codifica un termine } (n = \ulcorner t \urcorner) \\ 0 & \text{altrimenti.} \end{cases}$

Se $T(n)$ è primitivo ricorsivo, si ha la tesi

$$T(n) = 1 \iff n = \ulcorner 0 \urcorner \vee (\exists i \le n\ n = (\#(v), i)) \vee (\exists a, b < n\ \ n(\#(+), a, b) \wedge T(a) = 1 \wedge T(b) = 1)$$
$$\vee (\exists a, b < n\ \ n = (\#(\cdot), a, b) \wedge T(a) = 1 \wedge T(b) = 1)$$
$$\vee (\exists a < n\ \ n = (\#(S), a) \wedge T(a) = 1)$$

Questa può essere abbreviata come

$$T(n) = 1 \iff \exists a, b < n\ [\dots T(a) \dots T(b) \dots] \Rightarrow$$

$$\Rightarrow T(u) = h(u, \underbrace{(T(0), \dots, T(u-1))}_{S}), \text{ con } h(n, S) = 1 \iff$$

$$\iff \exists a, b < n\ [\dots (S)_a, \dots, (S)_b, \dots]$$

Teorema $\{ \ulcorner \varphi \urcorner \mid \varphi \text{ è } \mathcal{L}\text{-formula} \}$ è primitivo ricorsivo

OSS $\ulcorner \ \urcorner$ sulle formule è definita come

$$\ulcorner t_1 = t_2 \urcorner = (\#(=), \ulcorner t_1 \urcorner, \ulcorner t_2 \urcorner)$$

$$\ulcorner \neg \varphi \urcorner = (\#(\neg), \ulcorner \varphi \urcorner)$$

$$\ulcorner \alpha \wedge \beta \urcorner = (\#(\wedge) \ulcorner \alpha \urcorner, \ulcorner \beta \urcorner)$$

$$\ulcorner \forall x_i \varphi \urcorner = (\#(\forall), i, \ulcorner \varphi \urcorner)$$

$\vdots$

Dim. teo

Si definisca $F(n) = \begin{cases} 1 & n \text{ codifica la formula} \\ 0 & \text{altrimenti} \end{cases}$

$$F(u) = 1 \Leftrightarrow (\exists u, v < n \quad n = (\#(=), u, v) \wedge T(u) = 1 \wedge T(v) = 1) \vee$$

$$\vee (\exists u < n \quad n = (\#(\neg), u) \wedge T(u) = 1) \vee$$

$$\vee (\exists a, b < n \quad n = (\#(\wedge), a, b) \wedge F(a) = 1 = F(b)) \vee$$

$$\vee (\exists a < n, i < n \quad n = (\#(\forall), i, a) \wedge F(a) = 1)$$

OSS Trovare una codifica per gli assiomi di Q sarebbe facile, in quanto sono finiti

Se volessi codificare gli assiomi di Peano?

Si definisca induttivamente, intanto, la sostituzione $\varphi[t/x]$

Def
- $(\alpha \wedge \beta)[t/x] = \alpha[t/x] \wedge \beta[t/x]$
- $(\forall y\, \varphi)[t/x] = \forall y\, (\varphi[t/x])$
- $(\forall x\, \varphi)[t/x] = \forall x\, \varphi$
- $(\neg \varphi)[t/x] = \neg(\varphi[t/x])$
- $(t_1 = t_2)[t/x] = (t_1[t/x] = t_2[t/x])$
- $x[t/x] = t$
- $S(t_1)[t/x] = S(t_1[t/x])$

$\vdots$

Esiste $sub: \mathbb{N}^3 \to \mathbb{N}$ primitivo ricorsivo tale che $sub(\ulcorner \varphi \urcorner, i, \ulcorner t \urcorner) = \ulcorner \varphi[t/x_i] \urcorner$

$$sub(a, b, c) = \begin{cases} \text{se } a = (\#(\wedge), e, f) & sub(a,b,c) = (\#(\wedge), sub(e,b,c), sub(f,b,c)) \\ \vdots \end{cases}$$

<u>Teorema</u> "n codifica un assioma di induzione" è prim ric.

<u>Dim</u> $x = x_i$

$$n = \ulcorner \{ \overbrace{\varphi[{}^0/_x]}^{a} \wedge \forall x [\overbrace{\varphi}^{b} \to \overbrace{\varphi[{}^{S(x)}/_x]}^{c}] \} \to \forall x\, \varphi \urcorner$$

$$\exists a, b, c < n \quad n = (\#(\to), (\#(\wedge), a, (\#(\forall), i, (\#(\to), b, c))), (\#(\forall), i, b)).$$

$$\wedge\ F(b) = 1 \ \wedge\ a = sub(b, i, \ulcorner 0 \urcorner) \ \wedge\ c = sub(b, i, \ulcorner S(x_i) \urcorner)$$

<u>oss</u> In realtà andrebbe codificato considerando anche dei parametri iniziali $\forall y_1, \dots, y_n$ all'inizio, ma vabbè, sono cose che si fanno.

4/12/2017

BERARDUCCI
TEORIA

Si ricordi che, tramite la codifica di Godel, si erano inoltre definite due funzioni primitive ricorsive, cioè

- $num : \mathbb{N} \to \mathbb{N}$ tale che $n \mapsto \ulcorner S^n(0) \urcorner$
- $sub : \mathbb{N}^3 \to \mathbb{N}$ tale che $(\ulcorner \varphi \urcorner, i, \ulcorner t \urcorner) \mapsto \ulcorner \varphi[t/x_i] \urcorner$

OSS Il fatto che queste funzioni siano primitive ricorsive è conveniente in quanto, ricordando il fatto che se $f : \mathbb{N} \to \mathbb{N}$ ricorsiva totale esiste $\varphi_f(x,y)$ $\mathcal{L}$-formula tale che $\forall a,b \in \mathbb{N}$ $f(a) = b \Rightarrow Q \vdash \varphi_f(\bar{a}, \bar{b})$ e $f(a) \neq b \Rightarrow Q \vdash \neg\varphi_f(\bar{a}, \bar{b})$, è possibile ricondursi a studiare le cose in Q.

OSS In realtà oltre alla binumerabilità in Q (cioè la proprietà appena ricordata) servirebbe anche la proprietà:

- $f(a) = b \Rightarrow Q \vdash \forall y\, \varphi_f(\bar{a}, y) \leftrightarrow y = \bar{b}$

cioè che le funzioni siano binumerabili funzionalmente. (proprietà che in realtà mi implica le altre)

LEMMA DI DIAGONALIZZAZIONE

Lemma Dato $\varphi(x)$ esiste θ chiusa tale che $Q \vdash [\varphi(\overline{\ulcorner \theta \urcorner}) \leftrightarrow \theta]$

OSS Quindi in particolare vale anche in $\mathbb{N}$ $(\mathbb{N} \models [\varphi(\ulcorner \theta \urcorner) \leftrightarrow \theta])$

Esempio Sia $\mathcal{L} = \{0, S, +, \cdot\}$ e $\varphi(x)$ una $\mathcal{L}$-formula. Ha senso parlare di $\varphi(x^y)$?

Così direttamente no.
Sia però $\varphi_{exp}(x,y,z)$ tale che binumera funzionalmente $(a,b) \mapsto a^b$ cioè

$$a^b = c \Leftrightarrow Q \vdash [\forall z\, [\varphi_{exp}(\bar{a}, \bar{b}, z) \leftrightarrow z = \bar{c}]$$

Si pongo, dunque, $\varphi(x^y) := \forall z\, (\underbrace{z = x^y}_{\varphi_{exp}(x,y,z)} \to \varphi(z))$

Lemma Esiste $D : \mathbb{N} \to \mathbb{N}$ primitiva ricorsiva tale che per ogni $\mathcal{L}$-formula α $D(\ulcorner \alpha \urcorner) = \ulcorner \alpha(\overline{\ulcorner \alpha \urcorner}) \urcorner$, dove $\alpha(t) = \alpha[t/x_0]$.

Dim

$D(x) = sub(x, 0, num(x))$. Si ha che

$$D(\ulcorner \alpha \urcorner) = sub(\ulcorner \alpha \urcorner, 0, num(\ulcorner \alpha \urcorner)) = sub(\ulcorner \alpha \urcorner, 0, \underset{\ulcorner \overline{\ulcorner \alpha \urcorner} \urcorner}{\underset{\parallel}{\ulcorner S^{\ulcorner \alpha \urcorner}(0) \urcorner}}) =$$

ma, visto che $sub(\ulcorner \alpha \urcorner, 0, \ulcorner t \urcorner) = \ulcorner \alpha(t) \urcorner$ si ha

$$D(\ulcorner \alpha \urcorner) = \ulcorner \alpha(\overline{\ulcorner \alpha \urcorner}) \urcorner$$

OSS Questa funzione è ovviamente calcolabile

idea dim lemma di diag

Si cerca $\Theta = \alpha(\ulcorner\alpha\urcorner)$

Si vuole $Q \vdash \alpha(\ulcorner\alpha\urcorner) \leftrightarrow \varphi(\alpha(\ulcorner\alpha\urcorner))$

Si prende $\alpha(x) = \varphi(D(x))$

Dim

Sia $\delta(x,y)$ che binumera funzionalmente D (in Q)

$$D(a) = b \Rightarrow Q \vdash \forall y\,[\delta(\bar{a}, y) \leftrightarrow y = \bar{b}]$$

$$\varphi(D(x)) = \forall y\,(y = D(x) \to \varphi(y)) = \forall y\,(\delta(x,y) \to \varphi(y))$$

Si definisca $\alpha(x) = \forall y\,(\delta(x,y) \to \varphi(y))$ dove $x = x_0$

e si consideri $n = \ulcorner\alpha(x)\urcorner$

Si ponga $\Theta = \alpha(\bar{n}) = \alpha(\ulcorner\alpha\urcorner)$

potrebbero essere anche numeri non standard

In Q si ha $\Theta \equiv \alpha(\ulcorner\alpha\urcorner) \equiv \forall y\,(\delta(\ulcorner\alpha\urcorner, y) \to \varphi(y)) \Leftrightarrow$

$\Leftrightarrow \forall y\,[y = \ulcorner\alpha(\ulcorner\alpha\urcorner)\urcorner \to \varphi(y)] \Leftrightarrow \varphi(\ulcorner\alpha(\ulcorner\alpha\urcorner)\urcorner) \equiv \varphi(\ulcorner\Theta\urcorner)$

visto che δ binumerava funzionalmente

TEOREMA DI TARSKI DELL'INDEFINIBILITÀ DELLA VERITÀ

Teorema Sia $True \subseteq \mathbb{N}$, $True = \{\ulcorner\varphi\urcorner \mid \mathbb{N} \models \varphi\}$. $\forall n\ True \notin \Sigma_n^0$ ($\Rightarrow$ non decidibile, semidecidibile ecc.)

Dim

Se $True \in \Sigma_n^0$, allora esisterebbe $\varphi(x)$ $\mathcal{L}$-formula tale che $n \in True \Leftrightarrow \mathbb{N} \models \varphi(\bar{n})$

Sia β tale che $\mathbb{N} \models \beta \leftrightarrow \neg\varphi(\ulcorner\beta\urcorner)$, cioè $\mathbb{N} \models \beta \Leftrightarrow \mathbb{N} \models \neg\varphi(\ulcorner\beta\urcorner)$
(visto che sono in $\mathbb{N}$).
Ma

$$\mathbb{N} \models \neg\varphi(\ulcorner\beta\urcorner) \Leftrightarrow \ulcorner\beta\urcorner \notin True \Leftrightarrow \mathbb{N} \not\models \beta$$, assurdo.

oss Dopo aver codificato le formule e i termini, anche se non si dimostrerà, è possibile dare una codifica anche per le dimostrazioni.

oss Sarebbe dimostrabile (meccanicamente, ma è molto lungo da fare) che $Prov_{PA} = \{(n, \ulcorner\varphi\urcorner) \mid n$ codifica una dimostrazione di φ in $PA\}$ è primitivo ricorsivo, quindi è binumerato da $P(x,y)$ in Q

oss $Teo(y) \equiv \exists x\, P(x,y)$

Quindi $\{\ulcorner\varphi\urcorner \mid PA \vdash \varphi\} = \{\ulcorner\varphi\urcorner \mid \mathbb{N} \models \exists x\, P(x, \ulcorner\varphi\urcorner)\} =$

$= \{\ulcorner\varphi\urcorner \mid \exists x \in \mathbb{N}\ Prov_{PA}(x, \ulcorner\varphi\urcorner)\}$ è ric. enumerabile

in quanto $PA \vdash \varphi \Leftrightarrow \exists n\ (n, \ulcorner\varphi\urcorner) \in Prov_{PA} \Leftrightarrow \exists n\ Q \vdash P(\bar{n}, \ulcorner\varphi\urcorner) \Leftrightarrow$
$\Leftrightarrow Q \vdash \exists x\, P(x, \ulcorner\varphi\urcorner) \Leftrightarrow Q \vdash Teo(\ulcorner\varphi\urcorner)$

PRIMO TEOREMA DI GÖDEL

Teorema PA è incompleta.

Idea dim

Si cerca una formula G tale che $PA \nvdash G$, $PA \nvdash \neg G$,
cioè G tale che $Q \vdash G \Leftrightarrow \neg Teo(\ulcorner G \urcorner)$

Dim

Si ha che $PA \nvdash G$

Se $PA \vdash G$ allora $\exists n \; (n, \ulcorner G \urcorner) \in Prov \Rightarrow Q \vdash P(\bar{n}, \ulcorner G \urcorner) \Leftrightarrow$
$\Leftrightarrow Q \vdash \exists x \; P(x, \ulcorner G \urcorner) \Leftrightarrow Q \vdash Teo(\ulcorner G \urcorner) \Leftrightarrow Q \vdash \neg G \Rightarrow PA \vdash \neg G$
Ma PA è coerente, quindi si ha un assurdo

Si ha, inoltre, che $\mathbb{N} \models G$ se prov è bnum. in Q lo è anche in $\mathbb{N}$

Visto che $PA \nvdash G$, $\neg \exists n$ t.c. $(n, \ulcorner G \urcorner) \in Prov \Leftrightarrow \neg \exists n \; \mathbb{N} \models P(\bar{n}, \ulcorner G \urcorner) \Leftrightarrow$

$\Leftrightarrow \mathbb{N} \models \neg \exists x \; P(x, \ulcorner G \urcorner) \Leftrightarrow \mathbb{N} \models \neg Teo(\ulcorner G \urcorner) \Leftrightarrow \mathbb{N} \models G$

Perciò, visto che $\mathbb{N} \models G$, $PA \nvdash G$.
Quindi si ha una formula indecidibile che è vera.

OSS Questa dimostrazione non può essere fatta da Peano solo perché 'Peano non sa di essere coerente'.

Def Una teoria T è ω-coerente (in $\mathcal{L} = \{0, S, +, \cdot\}$) se non esiste una formula $\varphi(x)$ tale che $T \vdash \exists x \; \varphi(x)$, ma $T \vdash \neg\varphi(\bar{0})$, $T \vdash \neg\varphi(\bar{1})$, ..., $T \vdash \neg\varphi(\bar{n})$ $\forall n \in \mathbb{N}$

Esempio Sia $T = PA \cup \{\exists x \; P_{PA}(x, \ulcorner G \urcorner)\}$. Questa è coerente ma non ω-coerente.
$Teo(\ulcorner G \urcorner)$

Infatti $T \vdash \exists x \; P_{PA}(x, \ulcorner G \urcorner)$ in quanto è un assioma, ma $T \vdash \neg P(\bar{n}, \ulcorner G \urcorner)$ $\forall n$ (in quanto $Q \vdash \neg P(\bar{n}, \ulcorner G \urcorner)$ $\forall n$). Perciò non è ω-coerente.
Se fosse incoerente, si avrebbe $PA \vdash \neg Teo(G) \Rightarrow PA \vdash G$ ma si è visto che non è possibile.
Perciò T è coerente.