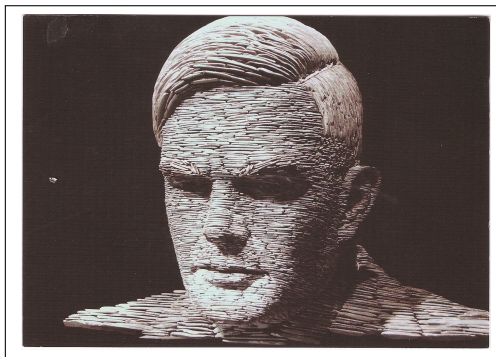


# *Clausole di Horn e computabilità*

Eugenio G. Omodeo Trieste, 18/05/2016



UNIVERSITÀ  
DEGLI STUDI DI TRIESTE



( Alan Mathison Turing, 1912–1954 )



UNIVERSITÀ  
DEGLI STUDI DI TRIESTE

- Di che si occupa la teoria della computabilità
- 
- 
- 



- Di che si occupa la teoria della computabilità
- Un'esplicitazione del concetto di funzione calcolabile
- 
- 



- Di che si occupa la teoria della computabilità
- Un'esplicitazione del concetto di funzione calcolabile
- Completezza di Turing della programmaz. tramite clausole di Horn
- 



- Di che si occupa la teoria della computabilità
- Un'esplicitazione del concetto di funzione calcolabile
- Completezza di Turing della programmaz. tramite clausole di Horn
- Indecidibilità della logica predicativa del 1° ordine





Di che si occupa  
la computabilità ?

( Mair Eluned Thomas Russell-Jones, 1917–2013 )



UNIVERSITÀ  
DEGLI STUDI DI TRIESTE

*“Computability theory [· · ·] is the theory of computation obtained when limitations of space and time are deliberately ignored.”*

*[Davis et al.(1994)Davis, Sigal, and Weyuker, pag. 237]*



*“Computability theory [· · ·] is the theory of computation obtained when limitations of space and time are deliberately ignored.”*

*[Davis et al.(1994)Davis, Sigal, and Weyuker, pag. 237]*

*“Computability theory ( also called recursion theory ) studies the class of partially computable functions.”*

*[Davis et al.(1994)Davis, Sigal, and Weyuker, pag. 31]*





*“Computability theory [· · ·] is the theory of computation obtained when limitations of space and time are deliberately ignored.”*

*[Davis et al.(1994)Davis, Sigal, and Weyuker, pag. 237]*

*“Computability theory ( also called recursion theory ) studies the class of partially computable functions.”*

*[Davis et al.(1994)Davis, Sigal, and Weyuker, pag. 31]*





WIKIPEDIA  
The Free Encyclopedia

## ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM

By **A. M. TURING**

[Received 28 May, 1936.—Read 12 November, 1936.]

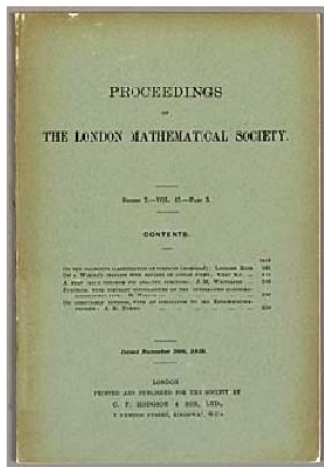
1. Computing machines.
2. Definitions.

*Automatic machines.*  
*Computing machines.*  
*Circle and circle-free numbers.*  
*Computable sequences and numbers.*

3. Examples of computing machines.
4. Abbreviated tables

*Further examples.*

5. Enumeration of computable sequences.
6. The universal computing machine.
7. Detailed description of the universal machine.
8. Application of the diagonal process.
9. The extent of the computable numbers.
10. Examples of large classes of numbers which are computable.
11. Application to the Entscheidungsproblem.



Quando una funzione  $g$  è definita su un sottoinsieme di  $\mathbb{N}^a$  e assume valori in  $\mathbb{N}$ , ciò viene indicato così:

$$g : \mathbb{N}^a \longrightarrow \mathbb{N}.$$

Diciamo che tale funzione è *parziale*, di *arietà*  $a$ .



Quando una funzione  $g$  è definita su un sottoinsieme di  $\mathbb{N}^a$  e assume valori in  $\mathbb{N}$ , ciò viene indicato così:

$$g : \mathbb{N}^a \rightarrow \mathbb{N}.$$

Diciamo che tale funzione è *parziale*, di *arietà*  $a$ .

**Esempio:** Qualsiasi polinomio

$$D \in \mathbb{Z}[k_1, \dots, k_a]$$

induce una funzione parziale  $g$  sulle  $a$ -uple

$$\langle n_1, \dots, n_a \rangle \in \mathbb{N}^a$$

secondo la regola:



Quando una funzione  $g$  è definita su un sottoinsieme di  $\mathbb{N}^a$  e assume valori in  $\mathbb{N}$ , ciò viene indicato così:

$$g : \mathbb{N}^a \longrightarrow \mathbb{N}.$$

Diciamo che tale funzione è *parziale*, di *arit*  $a$ .

**Esempio:** Qualsiasi polinomio

$$D \in \mathbb{Z}[k_1, \dots, k_a]$$

induce una funzione parziale  $g$  sulle  $a$ -uple

$$\langle n_1, \dots, n_a \rangle \in \mathbb{N}^a$$

secondo la regola:

$$g(n_1, \dots, n_a) = \begin{cases} D(n_1, \dots, n_a) & \text{purché questo val. non sia negativo,} \\ \text{non definito} & \text{se invece è negativo} \end{cases}$$



Quando—come nel caso dei polinomi menzionato ora—i valori di una funzione sono *calcolabili*, al termine '*non definito*' possiamo attribuire il significato di un comportamento perpetuo. . .



Quando—come nel caso dei polinomi menzionato ora—i valori di una funzione sono *calcolabili*, al termine ‘*non definito*’ possiamo attribuire il significato di un comportamento perpetuo. . .

. . . non sempre desiderabile o intenzionale !



Quando—come nel caso dei polinomi menzionato ora—i valori di una funzione sono *calcolabili*, al termine '*non definito*' possiamo attribuire il significato di un comportamento perpetuo. . .

. . . non sempre desiderabile o intenzionale !

; )





# OCCORRE FAR DI NECESSITÀ VIRTÙ ?

Generalizzando l'esempio di prima: Sia

$$D( \underbrace{k_1, \dots, k_a}_{\text{coefficients}}, \underbrace{x_1, \dots, x_m}_{\text{indeterminates}} )$$

un pol. a coefficienti in  $\mathbb{Z}$  nelle indeterminate



# OCCORRE FAR DI NECESSITÀ VIRTÙ ?

Generalizzando l'esempio di prima: Sia

$$D( \underbrace{k_1, \dots, k_a}_{\substack{\text{parametri} \\ \text{formali}}} , \underbrace{x_1, \dots, x_m}_{\text{incognite}} )$$

un pol. a coefficienti in  $\mathbb{Z}$  nelle indeterminate  $k_i$  e  $x_j$ .



# OCCORRE FAR DI NECESSITÀ VIRTÙ ?

Generalizzando l'esempio di prima: Sia

$$D(\underbrace{k_1, \dots, k_a}_{\text{parametri formali}}, \underbrace{x_1, \dots, x_m}_{\text{incognite}})$$

un pol. a coefficienti in  $\mathbb{Z}$  nelle indeterminate  $k_i$  e  $x_j$ .

Quali sono le  $a$ -uple

$$\langle n_1, \dots, n_a \rangle \in \mathbb{N}^a$$

per le quali l'eq.

$$D(\underbrace{n_1, \dots, n_a}_{\text{parametri impostati}}, \underbrace{x_1, \dots, x_m}_{\text{incognite}}) = 0$$

ammette soluzione su  $\mathbb{N}$  ?



Almeno una delle seguenti 2 funzioni è calcolabile ?

$$\tilde{g}(n_1, \dots, n_a) = \begin{cases} 1 & \text{se } D(n_1, \dots, n_a, x_1, \dots, x_m) = 0 \\ & \text{ha sol. su } \mathbb{N}, \\ 0 & \text{altrimenti.} \end{cases}$$



Almeno una delle seguenti 2 funzioni è calcolabile ?

$$\tilde{g}(n_1, \dots, n_a) = \begin{cases} 1 & \text{se } D(n_1, \dots, n_a, x_1, \dots, x_m) = 0 \\ & \text{ha sol. su } \mathbb{N}, \\ 0 & \text{altrimenti.} \end{cases}$$

$$\hat{g}(n_1, \dots, n_a) = \begin{cases} 1 & \text{se } D(n_1, \dots, n_a, x_1, \dots, x_m) = 0 \\ & \text{ha sol. su } \mathbb{N}, \\ \text{indef.} & \text{altrimenti.} \end{cases}$$



Almeno una delle seguenti 2 funzioni è calcolabile ?

$$\tilde{g}(n_1, \dots, n_a) = \begin{cases} 1 & \text{se } D(n_1, \dots, n_a, x_1, \dots, x_m) = 0 \\ & \text{ha sol. su } \mathbb{N}, \\ 0 & \text{altrimenti.} \end{cases}$$

$$\hat{g}(n_1, \dots, n_a) = \begin{cases} 1 & \text{se } D(n_1, \dots, n_a, x_1, \dots, x_m) = 0 \\ & \text{ha sol. su } \mathbb{N}, \\ \text{indef.} & \text{altrimenti.} \end{cases}$$

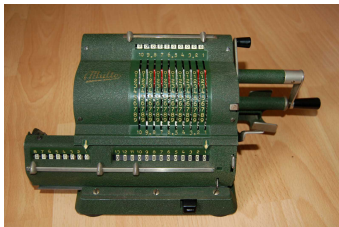
La  $\hat{g}$  sí, certo, ma la  $\tilde{g}$  chissà...



# UN *déjà vu* ?



# UN déjà vu ?







Quanto *superficiale* (?) è l'affinità fra l'enumerazione di tutte le dimostrazioni di una teoria e la ricerca sistematica degli zeri di un polinomio diofanteo?



*Hilbert*

David Hilbert avanzò 23 problemi in occasione  
del *Congresso internazionale dei matematici*  
a Parigi, l'8 agosto 1900





*Hilbert*

David Hilbert avanzò 23 problemi in occasione  
del *Congresso internazionale dei matematici*  
a Parigi, l'8 agosto 1900

**10. Entscheidung der Lösbarkeit einer diophantischen Gleichung** Eine diophantische Gleichung mit irgendwelchen Unbekannten und mit ganzen rationalen Zahlkoeffizienten sei vorgelegt: *Man soll ein Verfahren angeben, nach welchem sich mittels einer endlichen Anzahl von Operationen entscheiden läßt, ob die Gleichung in ganzen rationalen Zahlen lösbar ist.*



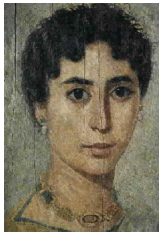
*Hilbert*

David Hilbert avanzò 23 problemi in occasione  
del *Congresso internazionale dei matematici*  
a Parigi, l'8 agosto 1900

## 10. Determinazione della risolubilità di un'equazione diofantea

Data un'equazione diofantea in qualsiasi numero d'incognite e a coefficienti interi razionali: *Ideare un procedimento per mezzo del quale si possa stabilire, in un numero finito di operazioni, se l'equazione sia o no risolubile negli interi razionali.*





David Hilbert avanzò 23 problemi in occasione  
del *Congresso internazionale dei matematici*  
a Parigi, l'8 agosto 1900

( Ipazia, ca. 370 / 415 d.C. )

## 10. Determinazione della risolubilità di un'equazione diofantea

Data un'equazione diofantea in qualsiasi numero d'incognite e a coefficienti interi razionali: *Ideare un procedimento per mezzo del quale si possa stabilire, in un numero finito di operazioni, se l'equazione sia o no risolubile negli interi razionali.*





Un'esplicitaz.  
del concetto di  
funzione calcolabile



UNIVERSITÀ  
DEGLI STUDI DI TRIESTE

**Basic Result, Part I** *By means of detailed combinatorial studies (see, for example, Turing [1937] and Kleene [1936a],) the proposed characterizations of Turing and of Kleene, as well as those of Church, Post, Markov, and certain others, were all shown to be equivalent; that is to say, exactly the same class of partial functions (and hence of total functions) is obtained in each case.*

**Definition** The functions falling within this class are called *recursive functions*. The partial functions of this class might, naturally, be termed “recursive partial functions.” It has become standard usage, however, to call them *partial recursive functions*.

*These equivalence demonstrations can be generalized to show that over certain very broad families of enlargements of these formal characterizations the class of partial functions obtained remains unchanged. (For example, if we allow*

[Rogers(1967), pag. 18]



Chiamiamo  $g$  *funzione parzialmente computabile* ( sui numeri naturali ) sse esiste una lista

$$g_0, \dots, g_M,$$

con  $M$  numero naturale qualsiasi e  $g_M = g$ , di funzioni

$$g_i : \mathbb{N}^{a_i} \longrightarrow \mathbb{N}$$

tali che ogni  $i = 0, \dots, M$  soddisfi almeno una delle sei condiz.







### *Funzioni iniziali:*

- (1)  $g_i$  è la funzione  $n \mapsto 0$  di arità  $a_i = 1$  che manda ogni numero nello 0;





### *Funzioni iniziali:*

- (1)  $g_0$  è la funzione  $n \mapsto 0$  di arità  $a_0 = 1$  che manda ogni numero nello 0;
- (2)  $g_1$  è la funzione  $n \mapsto n + 1$  di arità  $a_1 = 1$  che manda ciascun numero  $n$  nel suo successore immediato  $n + 1$ ;





### *Funzioni iniziali:*

- (1)  $g_i$  è la funzione  $n \mapsto 0$  di arità  $a_i = 1$  che manda ogni numero nello 0;
- (2)  $g_i$  è la funzione  $n \mapsto n + 1$  di arità  $a_i = 1$  che manda ciascun numero  $n$  nel suo successore immediato  $n + 1$ ;
- (3)  $g_i$  è la funzione  $\langle n_1, \dots, n_{a_i} \rangle \mapsto n_j$  di arità  $a_i > 0$  che manda—per qualche  $j$  con  $0 < j \leq a_i$ —ogni  $a_i$ -upla  $\langle n_1, \dots, n_{a_i} \rangle$  di numeri nella  $j$ -sima componente,  $n_j$ ;





(4)  $g_i$  è ottenuta per *composizione* da funzioni

$$g_h, g_{l_1}, \dots, g_{l_m}, \text{ con } h, l_1, \dots, l_m < i,$$

dove ciascuna delle  $g_{l_k}$  ha arità uguale all'arietà  $a_i$  di  $g_i$ , mentre  $g_h$  ha arità  $a_h = m$ .





(4)  $g_i$  è ottenuta per *composizione* da funzioni

$$g_h, g_{\ell_1}, \dots, g_{\ell_m}, \text{ con } h, \ell_1, \dots, \ell_m < i,$$

dove ciascuna delle  $g_{\ell_k}$  ha arità uguale all'arietà  $a_i$  di  $g_i$ , mentre  $g_h$  ha arità  $a_h = m$ .

Ciò significa che se in  $\langle n_1, \dots, n_{a_i} \rangle$  tutte le  $g_{\ell_k}$  sono definite, e definito è anche il valore

$$g_h \left( g_{\ell_1}(n_1, \dots, n_{a_i}), \dots, g_{\ell_m}(n_1, \dots, n_{a_i}) \right),$$

quest'ultimo verrà a coincidere con  $g_i(n_1, \dots, n_{a_i})$ ;





(4)  $g_i$  è ottenuta per *composizione* da funzioni

$$g_h, g_{l_1}, \dots, g_{l_m}, \text{ con } h, l_1, \dots, l_m < i,$$

dove ciascuna delle  $g_{l_k}$  ha arità uguale all'arietà  $a_i$  di  $g_i$ , mentre  $g_h$  ha arità  $a_h = m$ .

Ciò significa che se in  $\langle n_1, \dots, n_{a_i} \rangle$  tutte le  $g_{l_k}$  sono definite, e definito è anche il valore

$$g_h \left( g_{l_1}(n_1, \dots, n_{a_i}), \dots, g_{l_m}(n_1, \dots, n_{a_i}) \right),$$

quest'ultimo verrà a coincidere con  $g_i(n_1, \dots, n_{a_i})$ ; altrimenti  $g_i(n_1, \dots, n_{a_i})$  rimarrà *non definito*;





(5)  $g_i$  è ottenuta per *ricorsione primitiva* da funzioni  $g_h$  e  $g_\ell$ , con  $h, \ell < i$ , dove le arità  $a_i, a_h, a_\ell$  di  $i, h, \ell$  sono così legate fra loro:  $a_\ell = a_i + 1 = a_h + 2$ . Cioè a dire,  $g_i$  risulta definita dalla ricorrenza

$$\begin{cases} g_i(n_1, \dots, n_{a_h}, 0) & = g_h(n_1, \dots, n_{a_h}), \\ g_i(n_1, \dots, n_{a_h}, n+1) & = g_\ell\left(n_1, \dots, n_{a_h}, n, g_i(n_1, \dots, n_{a_h}, n)\right), \end{cases}$$





- (5)  $g_i$  è ottenuta per *ricorsione primitiva* da funzioni  $g_h$  e  $g_\ell$ , con  $h, \ell < i$ , dove le arità  $a_i, a_h, a_\ell$  di  $i, h, \ell$  sono così legate fra loro:  $a_\ell = a_i + 1 = a_h + 2$ . Cioè a dire,  $g_i$  risulta definita dalla ricorrenza

$$\begin{cases} g_i(n_1, \dots, n_{a_h}, 0) & = g_h(n_1, \dots, n_{a_h}), \\ g_i(n_1, \dots, n_{a_h}, n+1) & = g_\ell\left(n_1, \dots, n_{a_h}, n, g_i(n_1, \dots, n_{a_h}, n)\right), \end{cases}$$

che, per ogni  $a_h$ -upla  $\langle n_1, \dots, n_{a_h} \rangle$  di numeri, consente di determinare il valore  $g_i(n_1, \dots, n_{a_h}, 0)$  sse  $g_h(n_1, \dots, n_{a_h})$  è definito e, in tal caso, fornisce anche i valori  $g_i(n_1, \dots, n_{a_h}, m)$  per  $m = 1, 2, 3, \dots$ ,







- (5)  $g_i$  è ottenuta per *ricorsione primitiva* da funzioni  $g_h$  e  $g_\ell$ , con  $h, \ell < i$ , dove le arità  $a_i, a_h, a_\ell$  di  $i, h, \ell$  sono così legate fra loro:  $a_\ell = a_i + 1 = a_h + 2$ . Cioè a dire,  $g_i$  risulta definita dalla ricorrenza

$$\begin{cases} g_i(n_1, \dots, n_{a_h}, 0) & = g_h(n_1, \dots, n_{a_h}), \\ g_i(n_1, \dots, n_{a_h}, n+1) & = g_\ell\left(n_1, \dots, n_{a_h}, n, g_i(n_1, \dots, n_{a_h}, n)\right), \end{cases}$$

che, per ogni  $a_h$ -upla  $\langle n_1, \dots, n_{a_h} \rangle$  di numeri, consente di determinare il valore  $g_i(n_1, \dots, n_{a_h}, 0)$  sse  $g_h(n_1, \dots, n_{a_h})$  è definito e, in tal caso, fornisce anche i valori

$g_i(n_1, \dots, n_{a_h}, m)$  per  $m = 1, 2, 3, \dots$ ,

sin quando—se mai ciò accade—non risulti indefinito

$g_\ell\left(n_1, \dots, n_{a_h}, m-1, g_i(n_1, \dots, n_{a_h}, m-1)\right)$ ;





(6)  $g_i$  è ottenuta per *minimalizzazione* da una funzione  $g_\ell$ , con  $\ell < i$ , dove vige  $a_\ell = a_i + 1$  fra le arità di  $g_\ell$  e di  $g_i$ .

Indichiamo con  $\tilde{g}_i$  la funzione

$$\langle \vec{x}, m \rangle \xrightarrow{\tilde{g}_i} \min_{n \geq m} g_\ell(\vec{x}, n) = 0$$

di arità  $a_\ell$





(6)  $g_i$  è ottenuta per *minimalizzazione* da una funzione  $g_\ell$ , con  $\ell < i$ , dove vige  $a_\ell = a_i + 1$  fra le arità di  $g_\ell$  e di  $g_i$ .

Indichiamo con  $\tilde{g}_i$  la funzione

$$\langle \vec{x}, m \rangle \xrightarrow{\tilde{g}_i} \min_{n \geq m} g_\ell(\vec{x}, n) = 0$$

di arità  $a_\ell$  tale che  $\tilde{g}_i(n_1, \dots, n_{a_i}, m)$  è definita sse vi è un  $n \geq m$  tale che

- $g_\ell(n_1, \dots, n_{a_i}, n) = 0$  ed inoltre (se  $n > m$ )
- $g_\ell(n_1, \dots, n_{a_i}, m), g_\ell(n_1, \dots, n_{a_i}, m+1), \dots, g_\ell(n_1, \dots, n_{a_i}, n-1)$  sono tutti definiti e diversi da 0.





(6)  $g_i$  è ottenuta per *minimalizzazione* da una funzione  $g_\ell$ , con  $\ell < i$ , dove vige  $a_\ell = a_i + 1$  fra le arità di  $g_\ell$  e di  $g_i$ .

Indichiamo con  $\tilde{g}_i$  la funzione

$$\langle \vec{x}, m \rangle \xrightarrow{\tilde{g}_i} \min_{n \geq m} g_\ell(\vec{x}, n) = 0$$

di arità  $a_\ell$  tale che  $\tilde{g}_i(n_1, \dots, n_{a_i}, m)$  è definita sse vi è un  $n \geq m$  tale che

- $g_\ell(n_1, \dots, n_{a_i}, n) = 0$  ed inoltre (se  $n > m$ )
- $g_\ell(n_1, \dots, n_{a_i}, m), g_\ell(n_1, \dots, n_{a_i}, m+1), \dots, g_\ell(n_1, \dots, n_{a_i}, n-1)$  sono tutti definiti e diversi da 0.

Quando queste condizioni sono soddisfatte definiamo:

$$\tilde{g}_i(n_1, \dots, n_{a_i}, m) = n.$$





(6)  $g_i$  è ottenuta per *minimalizzazione* da una funzione  $g_\ell$ , con  $\ell < i$ , dove vige  $a_\ell = a_i + 1$  fra le arità di  $g_\ell$  e di  $g_i$ .

Indichiamo con  $\tilde{g}_i$  la funzione

$$\langle \vec{x}, m \rangle \xrightarrow{\tilde{g}_i} \min_{n \geq m} g_\ell(\vec{x}, n) = 0$$

di arità  $a_\ell$  tale che  $\tilde{g}_i(n_1, \dots, n_{a_i}, m)$  è definita sse vi è un  $n \geq m$  tale che

- $g_\ell(n_1, \dots, n_{a_i}, n) = 0$  ed inoltre (se  $n > m$ )
- $g_\ell(n_1, \dots, n_{a_i}, m), g_\ell(n_1, \dots, n_{a_i}, m+1), \dots, g_\ell(n_1, \dots, n_{a_i}, n-1)$  sono tutti definiti e diversi da 0.

Quando queste condizioni sono soddisfatte definiamo:

$$\tilde{g}_i(n_1, \dots, n_{a_i}, m) = n.$$

Si avrà allora, per come va intesa la minimalizzazione, che

$$g_i(n_1, \dots, n_{a_i}) = \tilde{g}_i(n_1, \dots, n_{a_i}, 0).$$



Supponendo che per ogni  $x$  in  $\mathbb{N}$  ed ogni  $y$  in  $\mathbb{N}$ :

- $g_0(x) = 0$  ,

queste funzioni sono parzialm. computabili ?



Supponendo che per ogni  $x$  in  $\mathbb{N}$  ed ogni  $y$  in  $\mathbb{N}$ :

- $g_0(x) = 0$  ,
- $g_1(x) = x + 1$  ,

queste funzioni sono parzialm. computabili ?



Supponendo che per ogni  $x$  in  $\mathbb{N}$  ed ogni  $y$  in  $\mathbb{N}$ :

- $g_0(x) = 0$  ,
- $g_1(x) = x + 1$  ,
- $g_2(x, y) = x$  ,

queste funzioni sono parzialm. computabili ?





Supponendo che per ogni  $x$  in  $\mathbb{N}$  ed ogni  $y$  in  $\mathbb{N}$ :

- $g_0(x) = 0$  ,
- $g_1(x) = x + 1$  ,
- $g_2(x, y) = x$  ,
- $g_3(x, y) = x + 1$  ,

queste funzioni sono parzialm. computabili ?



Supponendo che per ogni  $x$  in  $\mathbb{N}$  ed ogni  $y$  in  $\mathbb{N}$ :

- $g_0(x) = 0$  ,
- $g_1(x) = x + 1$  ,
- $g_2(x, y) = x$  ,
- $g_3(x, y) = x + 1$  ,
- $g_4(x)$  non sia definito ,

queste funzioni sono parzialm. computabili ?



Mostrare che sono parzialm. computabili le funzioni qui specificate tramite clausole di Horn:

$$\text{suc}(s(X), X);$$



Mostrare che sono parzialm. computabili le funzioni qui specificate tramite clausole di Horn:

$$\begin{aligned}
 & \text{suc}(s(X), X); \\
 & +(R, X, Y) \leftarrow \text{suc}(X, P) \quad \wedge \\
 & \quad \quad \quad +(Q, P, Y) \quad \wedge \\
 & \quad \quad \quad \text{suc}(R, Q), \\
 & +(Y, 0, Y);
 \end{aligned}$$



Mostrare che sono parzialm. computabili le funzioni qui specificate tramite clausole di Horn:

$$\begin{aligned}
 & \text{suc}(s(X), X); \\
 & +(R, X, Y) \leftarrow \text{suc}(X, P) \quad \wedge \\
 & \quad \quad \quad +(Q, P, Y) \quad \wedge \\
 & \quad \quad \quad \text{suc}(R, Q), \\
 & +(Y, 0, Y); \\
 & *(R, X, Y) \leftarrow +(X, A_1, A_2) \quad \wedge \\
 & \quad \quad \quad *(R_1, A_1, Y) \quad \wedge \\
 & \quad \quad \quad *(R_2, A_2, Y) \quad \wedge \\
 & \quad \quad \quad +(R, R_1, R_2), \\
 & *(0, 0, Y), \\
 & *(Y, s(0), Y).
 \end{aligned}$$



The claim that each of the standard formal characterizations provides satisfactory counterparts to the informal notions of *algorithm* and *algorithmic function* cannot be proved. It must be accepted or rejected on grounds that are, in large part, empirical. (That the claim for one charac-

see question \*10 in §1.1.) On the basis of this evidence, many mathematicians have accepted the claim that the standard characterizations give a satisfactory formalization, or “rational reconstruction,” of the (necessarily vague) informal notions. This claim is often referred to as *Church's Thesis*. Church's Thesis may be viewed as a *proposal* as well as a claim, a proposal that we agree henceforth to supply certain previously intuitive terms (e.g., “function computable by algorithm”) with certain precise meanings.

[Rogers(1967), pag. 20]





Completezza di Turing  
della programmaz.  
tramite clausole  
di Horn

( S.-Å. Tjärnlund, 1976/77 )  
( J. Šebelík & P. Štěpánek, 1982 )



UNIVERSITÀ  
DEGLI STUDI DI TRIESTE

# COME STABILIREMO LA COMPLETEZZA DI TURING

Per ogni lista di  $g_i$  del tipo su descritto, costruiremo basi

$$\emptyset = \mathcal{B}_0 \subset \mathcal{B}_1 \subset \mathcal{B}_2 \subset \dots \subset \mathcal{B}_{M+1}$$
$$g_0 \quad g_1 \quad g_2 \quad \dots \quad g_M$$

di clausole





# COME STABILIREMO LA COMPLETEZZA DI TURING

Per ogni lista di  $g_i$  del tipo su descritto, costruiremo basi

$$\emptyset = \mathcal{B}_0 \subset \mathcal{B}_1 \subset \mathcal{B}_2 \subset \dots \subset \mathcal{B}_{M+1}$$
$$g_0 \quad g_1 \quad g_2 \quad \dots \quad g_M$$

di clausole nella quale compariranno solo la costante  $0$ , il funtore monadico  $s(\_)$  di *successore* e letterali

$$p_i(t_0, \dots, t_{a_i}), \quad q_i(t_0, \dots, t_{a_i+1})$$

e relative negazioni, dove  $i = 0, 1, \dots, M$ .



# COME STABILIREMO LA COMPLETEZZA DI TURING

Per ogni lista di  $g_i$  del tipo su descritto, costruiremo basi

$$\emptyset = \mathcal{B}_0 \subset \mathcal{B}_1 \subset \mathcal{B}_2 \subset \dots \subset \mathcal{B}_{M+1}$$
$$g_0 \qquad g_1 \qquad g_2 \quad \dots \quad g_M$$

di clausole nella quale compariranno solo la costante  $0$ , il funtore monadico  $s(\_)$  di *successore* e letterali

$$p_i(t_0, \dots, t_{a_i}), \quad q_i(t_0, \dots, t_{a_i+1})$$

e relative negazioni, dove  $i = 0, 1, \dots, M$ .

Ciascun simbolo predicativo  $p_i$  sta a rappresentare, nelle nostre intenzioni, la funzione  $g_i$ , mentre il simbolo  $q_i$  svolge un ruolo ausiliario e rappresenta la funzione  $\tilde{g}_i$  quando  $g_i$  è definita per minimalizzazione.



# COME STABILIREMO LA COMPLETEZZA DI TURING

Per ogni lista di  $g_i$  del tipo su descritto, costruiremo basi

$$\emptyset = \mathcal{B}_0 \subset \underset{\mathcal{G}_0}{\mathcal{B}_1} \subset \underset{\mathcal{G}_1}{\mathcal{B}_2} \subset \dots \subset \underset{\mathcal{G}_M}{\mathcal{B}_{M+1}}$$

di clausole nella quale compariranno solo la costante  $0$ , il funtore monadico  $s(\_)$  di *successore* e letterali

$$p_i(t_0, \dots, t_{a_i}), \quad q_i(t_0, \dots, t_{a_i+1})$$

e relative negazioni, dove  $i = 0, 1, \dots, M$ .

Ciascun simbolo predicativo  $p_i$  sta a rappresentare, nelle nostre intenzioni, la funzione  $g_i$ , mentre il simbolo  $q_i$  svolge un ruolo ausiliario e rappresenta la funzione  $\tilde{g}_i$  quando  $g_i$  è definita per minimalizzazione.

Ogni  $\mathcal{B}_{i+1}$  verrà a specificare la  $g_i$ .



I termini così definiti:

$$\begin{aligned} \underline{0} & \stackrel{=_{\text{Def}}}{=} \mathbf{0}, \\ \underline{m+1} & \stackrel{=_{\text{Def}}}{=} \mathbf{s}(\underline{m}), \end{aligned}$$

designano in modo univoco tutti i numeri naturali. Formano un universo di Herbrand: l'universo dei *numerali* ( in base 1 ).



I termini così definiti:

$$\begin{aligned} \underline{0} &=_{\text{Def}} \mathbf{0}, \\ \underline{m+1} &=_{\text{Def}} \mathbf{s(m)}, \end{aligned}$$

designano in modo univoco tutti i numeri naturali. Formano un universo di Herbrand: l'universo dei *numerali* ( in base 1 ).

La nostra costruzione dei  $\mathcal{B}_i$  sarà tale che ogni domanda della forma  $\neg p_i(Y, \underline{n_1}, \dots, \underline{n_{a_i}})$  ammetta risposte corrette se e solo se  $\mathbf{g}_i(n_1, \dots, n_{a_i})$  è definito, in tal caso avendo come unica risposta la sostituzione

$$Y \mapsto \underline{\mathbf{g}_i(n_1, \dots, n_{a_i})}.$$



Indichiamo per ciascuno dei casi (1)–(6) contemplati piú su quali siano le clausole che formano

$$\mathcal{B}_{i+1} \setminus \mathcal{B}_i$$

( quando  $i$  ricade in piú di un caso, ci si regoli a piacere ):



Indichiamo per ciascuno dei casi (1)–(6) contemplati piú su quali siano le clausole che formano

$$\mathcal{B}_{i+1} \setminus \mathcal{B}_i$$

( quando  $i$  ricade in piú di un caso, ci si regoli a piacere ):

$$(1) p_i(\mathbf{0}, X);$$



Indichiamo per ciascuno dei casi (1)–(6) contemplati piú su quali siano le clausole che formano

$$\mathcal{B}_{i+1} \setminus \mathcal{B}_i$$

( quando  $i$  ricade in piú di un caso, ci si regoli a piacere ):

$$(1) \ p_i(\mathbf{0}, X);$$

$$(2) \ p_i(s(X), X);$$





Indichiamo per ciascuno dei casi (1)–(6) contemplati piú su quali siano le clausole che formano

$$\mathcal{B}_{i+1} \setminus \mathcal{B}_i$$

( quando  $i$  ricade in piú di un caso, ci si regoli a piacere ):

- (1)  $p_i(\mathbf{0}, X)$ ;
- (2)  $p_i(s(X), X)$ ;
- (3)  $p_i(X_j, X_1, \dots, X_{a_i})$ ;



$$(4) \quad p_i(Y, X_1, \dots, X_{a_i}) \leftarrow$$

$$p_h(Y, Y_{l_1}, \dots, Y_{l_{a_h}}) \quad \&$$

$$p_{l_1}(Y_{l_1}, X_1, \dots, X_{a_i}) \quad \&$$

...

...

&

$$p_{l_{a_h}}(Y_{l_{a_h}}, X_1, \dots, X_{a_i});$$



$$(4) \quad p_i(Y, X_1, \dots, X_{a_i}) \leftarrow$$

$$p_h(Y, Y_{l_1}, \dots, Y_{l_{a_h}}) \quad \&$$

$$p_{l_1}(Y_{l_1}, X_1, \dots, X_{a_i}) \quad \&$$

...

...

&

$$p_{l_{a_h}}(Y_{l_{a_h}}, X_1, \dots, X_{a_i});$$

$$(5) \quad p_i(Y, X_1, \dots, X_{a_h}, \mathbf{0}) \leftarrow p_h(Y, X_1, \dots, X_{a_h}),$$



$$(4) \quad p_i(Y, X_1, \dots, X_{a_i}) \leftarrow$$

$$p_h(Y, Y_{\ell_1}, \dots, Y_{\ell_{a_h}}) \quad \&$$

$$p_{\ell_1}(Y_{\ell_1}, X_1, \dots, X_{a_i}) \quad \&$$

...

...

&

$$p_{\ell_{a_h}}(Y_{\ell_{a_h}}, X_1, \dots, X_{a_i});$$

$$(5) \quad p_i(Y, X_1, \dots, X_{a_h}, \mathbf{0}) \leftarrow p_h(Y, X_1, \dots, X_{a_h}),$$

$$p_i(Y_\ell, X_1, \dots, X_{a_h}, s(X)) \leftarrow$$

$$p_\ell(Y_\ell, X_1, \dots, X_{a_h}, X, Y_i) \quad \&$$

$$p_i(Y_i, X_1, \dots, X_{a_h}, X);$$



$$(6) \quad p_i(Y, X_1, \dots, X_{a_i}) \leftarrow q_i(Y, X_1, \dots, X_{a_i}, \mathbf{0}),$$

—Fine della costruzione / dimostrazione—



$$(6) \quad p_i(Y, X_1, \dots, X_{a_i}) \leftarrow q_i(Y, X_1, \dots, X_{a_i}, \mathbf{0}),$$

$$q_i(Z, X_1, \dots, X_{a_i}, Z) \leftarrow p_i(\mathbf{0}, X_1, \dots, X_{a_i}, Z),$$

—Fine della costruzione / dimostrazione—



$$(6) \quad p_i(Y, X_1, \dots, X_{a_i}) \leftarrow q_i(Y, X_1, \dots, X_{a_i}, \mathbf{0}),$$

$$q_i(Z, X_1, \dots, X_{a_i}, Z) \leftarrow p_\ell(\mathbf{0}, X_1, \dots, X_{a_i}, Z),$$

$$q_i(Y_i, X_1, \dots, X_{a_i}, Z) \leftarrow$$

$$p_\ell(s(Y), X_1, \dots, X_{a_i}, Z) \& \\ q_i(Y_i, X_1, \dots, X_{a_i}, s(Z)).$$

—Fine della costruzione / dimostrazione—





# Indecidibilità della logica predicativa del 1° ordine



La vedremo ben presto



UNIVERSITÀ  
DEGLI STUDI DI TRIESTE





Martin D. Davis, Ron Sigal, and Elaine J. Weyuker.

*Computability, complexity, and languages - Fundamentals of theoretical computer science.*

Computer Science ad scientific computing. Academic Press, 1994.



John Wylie Lloyd.

*Foundations of Logic Programming.*

Symbolic Computation. Springer-Verlag, Berlin, 2<sup>nd</sup> edition, 1987.



Hartley Rogers, Jr.

*Theory of Recursive Functions and Effective Computability.*

Series in Higher Mathematics. McGraw-Hill, 1967.

