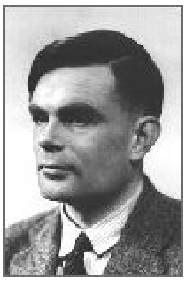


Alan Mathison Turing
(1912-1954)

Programma di massima del corso di

Logica 1

Antonio Greco

Anno accademico 2003/2004

Premesse. Equipotenza e numerabilità. Logica dei predicati del primo ordine. Assiomi. Teoremi. Regole di inferenza. Formule ben formate.

Computabilità effettiva. Insiemi ricorsivi. Insiemi ricorsivamente enumerabili. Nozione di funzione effettivamente computabile. Tesi di Church. Macchine di Turing. Esempi di problemi per i quali non può esistere un metodo risolutivo: il problema di stabilire se un enunciato è dimostrabile, il problema di stabilire se un polinomio a coefficienti interi ha radici intere, il problema di stabilire se una macchina di Turing si arresterà, il problema di stabilire se una funzione elementare è integrabile elementarmente.

Complessità computazionale. Efficienza di un algoritmo. Problemi intrattabili. Il problema del commesso viaggiatore. La scomposizione in fattori. Crittografia. La congettura $P=NP$.

Libro di testo:

- Toffalori, Cintioli, Logica Matematica, McGraw-Hill.

Finalità del corso. Fin da studente sono stato affascinato dal problema di stabilire a priori, ovvero di respingere, la *possibilità* di eseguire un certo calcolo. Nel corso cercherò di ricomporre le tessere di un mosaico che ho raccolto da diverse fonti in questi anni.

Durata del corso: 48 ore

Programma dettagliato del corso di

Logica 1

Antonio Greco

Anno accademico 2003/04

Premesse:

- Enunciato del teorema fondamentale dell'aritmetica (teorema di fattorizzazione in primi).

Equipotenza, cardinalità e numerabilità:

- Definizione di equipotenza [TC, pag. 1]. Definizione di numerabilità [TC, pag. 3].
- I procedimenti diagonali.
- Espressione, mediante le operazioni aritmetiche, di una corrispondenza biunivoca fra **N** e ciascuno dei seguenti insiemi:
  - l'insieme dei numeri dispari;
  - l'insieme dei numeri pari;
  - l'insieme dei quadrati perfetti (*paradosso di Galileo)*;
  - l'insieme **Z** [TC, pag. 2, esempio 1];
  - l'insieme **Z** privato di un intero *k*;
  - l'insieme **Z** privato di due interi *h,k*;
  - l'insieme $\mathbf{N}^2$ [TC, pag. 2, esempio 2].
- Esistenza di una corrispondenza biunivoca fra **N** e ciascuno dei seguenti insiemi:
  - l'insieme dei numeri primi;
  - l'insieme $\mathbf{N}^k$, con *k* intero positivo;
  - l'insieme **Q** [TC, pag. 2, esempio 4];
  - l'insieme S^* delle sequenze ordinate finite di elementi di un dato insieme finito *S*;
  - l'insieme $\mathbf{N}^*$;
  - l'insieme di quei numeri reali la cui rappresentazione decimale consta di un numero finito di cifre;
  - gli insiemi $\mathbf{N}[x]$, $\mathbf{Z}[x]$, $\mathbf{Q}[x]$;
  - un qualunque sottoinsieme infinito di **N**.
- Idea della dimostrazione delle seguenti proprietà:
  - l'unione di due insiemi numerabili è un insieme numerabile;
  - l'unione di un'infinità numerabile di insiemi numerabili è un insieme numerabile;
  - l'insieme *B* di tutti i sottoinsiemi finiti di un insieme numerabile *A* è un insieme numerabile.
- Non numerabilità dei seguenti insiemi:
  - l'insieme **R** [TC, pag. 3];
  - l'insieme $\mathbf{2}^{\mathbf{N}}$ [TC, pag. 3];
  - l'insieme $\mathbf{N}^{\mathbf{N}}$.
- Importanza dell'insieme $\mathbf{N}^{\mathbf{N}}$ nella teoria della computabilità.
- Esempi di funzioni iniettive da **N** verso $\mathbf{N}^{\mathbf{N}}$.

Esempi di problemi algoritmicamente risolubili:

- Calcolare la somma di due numeri interi.
- Dire se un dato numero è primo o composto.
- Determinare l'ennesimo numero primo.
- Scomporre in fattori un numero dato.
- Dire se un polinomio a coefficienti interi **e in una variabile** ha radici intere [TC, pag. 149, lettera (i) (piè di pagina)].

Funzioni ricorsive:

- Introduzione [TC, pag. 105].
- La funzione caratteristica di un sottoinsieme S di $\mathbf{N}$ [TC, pag. 108].
- Le funzioni ricorsive primitive e le funzioni ricorsive generali, la funzione di Ackermann (cenni) [TC, pag. 115, esempio 15].
- La funzione identità $\mathbf{1_N}$ è una funzione iniziale [TC, pag. 109].
- Ricorsività primitiva delle funzioni costanti [M, Corollario 3.14, lettera b], del fattoriale [M, Proposizione 3.15, lettera i], e delle funzioni somma, prodotto, elevamento a potenza, $pd(x)$, $sg(x)$, differenza aritmetica [TC, pagg. 112 e 113, esempi 1, 2, 3, 4, 5, 7].
- Descrizione di un algoritmo per effettuare l'operazione di minimalizzazione [TC, pag. 110].
- Funzioni parziali e funzioni totali [TC, pag. 108].
- Tesi di Church [TC, pag. 111].

Macchine di Turing:

- Definizione della macchina di Turing [TC, pag. 136].
- L'insieme delle macchine di Turing è numerabile (dispensa del docente).
- Rappresentazione *unaria* dei numeri naturali [TC, pag. 107].
- La computazione di una macchina di Turing: descrizione informale [TC, pag. 137].
- Concetto di funzione Turing-computabile [TC, Definizione a pag. 139].
- Esempi di macchine di Turing: calcolo della funzione successore, delle funzioni costanti, della funzione somma, delle proiezioni canoniche [TC, pag. 139, esempi 1, 2, 3, 4].
- Macchine di Turing che non si arrestano [TC, pag. 137, esempio].
- Macchine di Turing diverse tra loro ma che calcolano una stessa funzione (ad esempio, la funzione $succ(x)$).
- Uso di un simulatore della macchina di Turing (un simulatore è disponibile presso il docente).
- Equivalenza delle definizioni di algoritmo [TC, pag. 142].
- Aderenza delle definizioni all'intuizione: tesi di Turing [TC, pag. 141].
- Analogia tra la tesi di Turing e:
  - la definizione di limite;
  - l'interpretazione geometrica della derivata.

Gli algoritmi sono pochi rispetto alle funzioni:

- Dimostrazione, con considerazioni di cardinalità, dell'esistenza di funzioni non computabili [TC, Proposizione a pag. 111].
- Numeri di Gödel (funzione #) delle macchine di Turing [TC, pag. 146].

Insiemi ricorsivi e insiemi ricorsivamente enumerabili:

- Definizione di *insieme ricorsivo* [TC, pagina 111].
- Definizione di *insieme ricorsivamente enumerabile* [TC, pag. 117].
- Un sottoinsieme X di $\mathbf{N}$ è ricorsivo se e solo se X e $\mathbf{N} \setminus X$ sono ricorsivamente enumerabili (giustificazione intuitiva) [TC, Teorema a pag. 117].
- Dimostrazione, con considerazioni di cardinalità, dell'esistenza di insiemi non ricorsivamente enumerabili.
- L'insieme delle funzioni totali ricorsive non è ricorsivamente enumerabile [BC, Teorema 2.10 del paragrafo 2.2.4].
- Esistenza di un insieme ricorsivamente enumerabile ma non ricorsivo [TC, pagina 146].

Altri esempi di problemi che non ammettono metodo risolutivo:

- Il decimo problema di Hilbert [TC, paragrafo 4.10].

Elementi di teoria della complessità:

- Generalità [TC, pagine 154 e 155].
- Confronto tra il metodo di Laplace ed il metodo di Gauss per il calcolo di un determinante [BC, esempio 3.5].
- Complessità c_M di una macchina di Turnig M [TC, pagina 155].
- Il simbolo O ("o grande") [TC, pagina 156].
- La classe P [TC, pagina 158] e la classe NP [TC, pagina161].
- Il problema del commesso viaggiatore [TC, pagina 170, punto 2].
- $P = NP$? [TC, pagina 163].

L'**esame** consiste in una prova orale. Non è prevista una prova scritta. Al momento dell'esame, lo studente deve dimostrare di conoscere gli argomenti di questo programma, e di saper risolvere degli esercizi dello stesso livello di difficoltà di quelli assegnati durante il corso. Gli esercizi del corso sono reperibili presso il docente.

I riferimenti tra parentesi quadre corrispondono alla seguente **bibliografia:**

[BC] A. Bernasconi, B. Codenotti, Introduzione alla complessità computazionale, Springer-Verlag Italia, 1998 (AMS 03 D 15 1, AMS 03 D 15 2).

[M] E. Mendelson, Introduzione alla logica matematica, Boringhieri, 1987 (500.9, AMS 03 3, AMS 03 4).

[TC] C. Toffalori, P. Cintioli, Logica Matematica, McGraw-Hill, 2000.

(Le sigle fra parentesi tonde sono gli estremi della collocazione nella Biblioteca di Matematica)

Bibliografia

(Fra parentesi tonde sono riportati, quando possibile, gli estremi della collocazione nella Biblioteca di Matematica)

Testi di base

- Bencivenga, *Il primo libro di logica*, Boringhieri 1984.
- Bernasconi, Codenotti, *Introduzione alla complessità computazionale*, Springer-Verlag Italia 1998 (AMS 03 D 15 1, AMS 03 D 15 2).
- Consiglio del Corso di Laurea in Matematica dell'Università di Roma "Tor Vergata" (a cura del), *Matematica Zero, 30 definizioni alla base del linguaggio matematico*, 1991.
- Lolli, *Lezioni di logica matematica*, Boringhieri 1978.
- Mendelson, *Introduzione alla logica matematica*, Boringhieri 1972 (500.9, AMS 03 3, AMS 03 4).
- Shoenfield, *Logica matematica*, Boringhieri 1980 (Lo IV 32).
- Toffalori, Cintioli, *Logica matematica*, McGraw-Hill Italia 2000.

Crittografia

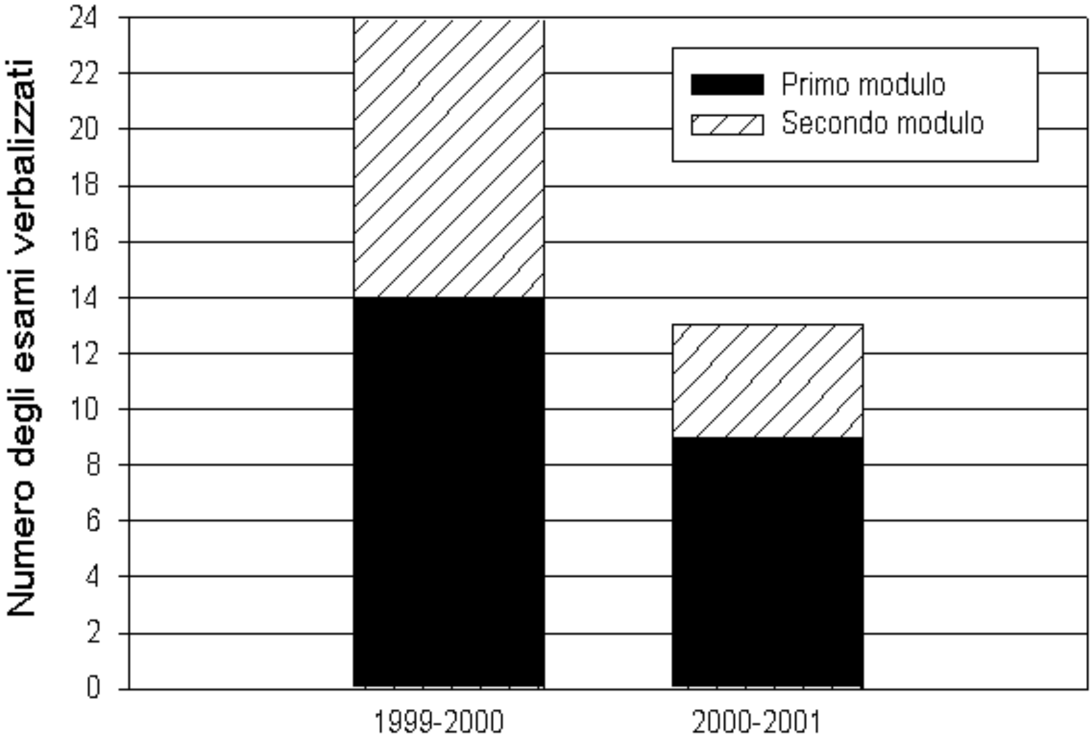
- Berardi, Beutelspacher, *Come rendere sicura la posta elettronica*. Archimede **51**(3) (1999), 137–143.
- Ferragina, Luccio, *Crittografia*, Boringhieri 2001.
- Giustozzi, Monti, Zimuel, *Segreti spie codici cifrati*, Apogeo 1999 (AMS 68-01 30).
- Massidda, *Metodi matematici della moderna crittografia*, tesi di laurea in Matematica, Cagliari 11-12-2000.
- Renzoni *Test probabilistici di primalità*, Archimede **54**(3) (2002), 146–149.
- Salomaa, *Public-key cryptography*, Second edition, Springer-Verlag 1996 (AMS 68P25 1).
- *La cryptographie expliquée*, http://www.bibmath.net/crypto/

Approfondimenti

- Barozzi, *Teoria elementare dei numeri con la TI-89/92*, Ipotesi, anno 2, n. 2, maggio 1999.
- Barwise, *How to tell if you are in the realm of fist-order logic*, Handbook of mathematical logic, North-Holland 1989, 6–17 (Lo IV 37).
- Boolos, Jeffrey, *Computability and logic*, Cambridge 1974 (Lo IV 21).
- Bridges, *Computability: a mathematical sketchbook*, Springer-Verlag 1994 (Coll. XXXVI 146).
- Brogi, Cisternino, Romani, *Le Macchine di Turing*, ipertesto del Dipartimento di Informatica dell'Università di Pisa.
- Casari, *Computabilità e ricorsività*, Quaderno n. **3** della Scuola di Studi Superiori sugli Idrocarburi dell'E.N.I. (1959) (Lo II 4).
- Casula, *Il decimo problema di Hilbert*, tesi di laurea in Matematica, Cagliari 20-12-2001.
- Davis, *Computability and unsolvability*, McGraw-Hill 1958 (Lo II 3).
- Davis, *Hilbert's tenth problem is unsolvable*, Am. Math. Mon. **80** (1973), 233–269.
- Davis (a cura di), *The undecidable*. Raven Press 1965 (510 UND).
- Greco, *Integrazione in termini finiti*, Archimede **51**(2) (1999), 82–88.
- Hermes, *Enumerability decidability computability*, Springer-Verlag 1965 (Coll. VIII 98), trad. it. *Enumerabilità decidibilità computabilità*, Boringhieri 1975 (4.D.785/9 nella Biblioteca Universitaria, via Università, Cagliari).
- Hopcroft, Ullman, *Introduction to automata theory, languages and computation*, Addison Wesley 1979 (AMS 68Q 1).
- Lolli, *Definizioni di algoritmo*, Le Scienze Quaderni 14 (1984), 21–25.
- Richardson, *Some undecidable problems involving elementary functions of a real variable*, J. Symb. Logic **33** (1968), 514–520.

Dati statistici sull'esame di

Logica Matematica



1999-2000

Primo modulo:
voto medio: 29,0
voto massimo: 30
voto minimo: 26
lodi: 2

Secondo modulo:
voto medio: 27,7
voto massimo: 30
voto minimo: 23
lodi: 0

2000-2001

Primo modulo:
voto medio: 28,7
voto massimo: 30
voto minimo: 22
lodi: 4

Secondo modulo:
voto medio: 27,0
voto massimo: 30
voto minimo: 23
lodi: 0

Antonio Greco - 4/3/02

Logica 1
a.a. 2003/04
N. 12

Esercizi

1) Per ogni coppia di funzioni $f, g \in \mathbb{N}^{\mathbb{N}}$, diciamo che f *è simile a* g, e scriviamo $f \sim g$, se $f = O(g)$ e $g = O(f)$. Dimostrare che la relazione $\sim$ appena definita è una relazione di equivalenza.

2) Fra le seguenti funzioni, individuare quelle simili tra loro.

$f_0(n) = 0$ per ogni $n \in \mathbb{N}$;

$f_1(n) = 1$ per ogni $n \in \mathbb{N}$;

$f_2(n) = (-1)^n + 1$ per ogni $n \in \mathbb{N}$;

$f_3(n) = n^3$ per ogni $n \in \mathbb{N}$;

$f_4(n) = n^3 + 50\,n^2 + 600\,n + 999$ per ogni $n \in \mathbb{N}$;

$f_6(n) = 6^n$ per ogni $n \in \mathbb{N}$.

3) Per ogni $f \in \mathbb{N}^{\mathbb{N}}$, definiamo l'insieme $f/\!\sim$ ponendo $f/\!\sim\; = \{\, h \in \mathbb{N}^{\mathbb{N}} \mid f \sim h \,\}$. Dimostrare che per ogni $g \in \mathbb{N}^{\mathbb{N}}$ si ha $f/\!\sim\; = g/\!\sim$ se e solo se $f \sim g$.

4) Per una generica $f \in \mathbb{R}^{\mathbb{N}}$, dimostrare che:

- se f ammette limite finito per $n \to +\infty$, allora esiste $c \in \mathbb{Z}^+$ tale che $f(n) \leq c$ per ogni $n \in \mathbb{N}$;
- se $f(n) \to 0$ per $n \to +\infty$, allora esiste $n_0 \in \mathbb{N}$ tale che per ogni $n > n_0$ risulta $f(n) < 1$.

Logica 1 a.a. 2003/04 N. 11

Esercizi

1) Per $n \in \mathbb{N}$ e $x \in \mathbb{Z}^+$, diciamo che n è *l'x-esimo numero di macchina* se $n \in \sharp(\mathcal{M})$, e l'insieme $\sharp(\mathcal{M}) \cap \{ k \mid k \leq n \}$ ha esattamente x elementi. Ideare un algoritmo per calcolare l'x-esimo numero di macchina, qualunque sia $x \in \mathbb{Z}^+$. Spunto: cominciare da $x = 1$.

2) Ideare un algoritmo che, per ogni macchina di Turing $M \in \mathcal{M}$, determini quell'intero x tale che $\sharp(M)$ è l'x-esimo numero di macchina.

3) Fare un esempio di funzione biunivoca e computabile $f\colon \mathcal{M} \to \mathbb{N}$.

4) Diciamo che una macchina di Turing $M \in \mathcal{M}$ è la *x-esima macchina* se $\sharp(M)$ è l'x-esimo numero di macchina. In tal caso, scriviamo $M = M_x$. Ideare un algoritmo per determinare M_x, qualunque sia $x \in \mathbb{Z}^+$.

5) Dico che l'insieme $K = \{ x \in \mathbb{N} \mid$ la computazione di M_x che inizia da $\lceil x \rceil$ termina$\}$ è ricorsivo, perché per ogni $x \in \mathbb{N}$ posso determinare la macchina M_x, e quindi applicarla all'input $\lceil x \rceil$ per vedere se la computazione termina. Dove sta l'errore?

6) Posso concludere che K non è ricorsivo?

Logica 1
a.a. 2003/04
N. 10

Esercizi

1) Si consideri la macchina di Turing le cui istruzioni sono: $(q_0, 1, 1, 1, q_1)$; $(q_1, *, *, 1, q_1)$. Stabilire se la computazione che parte dalla stringa $\lceil n \rceil$ termina, qualunque sia $n \in \mathbb{N}$.

2) Dimostrare che la funzione sg, così definita: $\mathrm{sg}(0) = 0$; $\mathrm{sg}(\mathrm{succ}\,(x)) = 1$ per $x \geq 0$, è ricorsiva primitiva.

3) Posto $A_k = \mathbb{Z}^+ + k = \{\, x \mid x > k \,\}$, dimostrare che la funzione $\chi_{A_k}(x) = \mathrm{sg}(x \dot{-} k)$ è ricorsiva primitiva qualunque sia $k \in \mathbb{N}$.

4) Dimostrare che la funzione $f_k \colon \mathbb{N} \to \mathbb{N}$, così definita:

$$f_k(x) = \begin{cases} x, & \text{se } x \leq k; \\ x + 1, & \text{se } x > k \end{cases}$$

è ricorsiva primitiva qualunque sia $k \in \mathbb{N}$.

5) Dimostrare che la funzione $\chi_{2\mathbb{N}}(x) = \mathrm{Re}(x, 2)$, così definita: $\chi_{2\mathbb{N}}(0) = 0$; $\chi_{2\mathbb{N}}(\mathrm{succ}(x)) = 1 \dot{-} \chi_{2\mathbb{N}}(x)$, è ricorsiva primitiva.

Logica 1 a.a. 2003/04 N. 8

Esercizi

1) Qual è il massimo numero di istruzioni che una macchina di Turing può avere?

2) Studiare la cardinalità dei seguenti insiemi.

   (a) L'insieme di tutti i file possibili nel formato di Word.

   (b) L'insieme di tutti i file, nel formato di Word, che possono essere salvati su di un comune floppy-disk.

3) Dimostrare che la funzione pd, così definita: $\mathrm{pd}\,(0) = 0$; $\mathrm{pd}\,(\mathrm{succ}\,(x)) = x$ per $x \geq 0$, è ricorsiva primitiva.

4) Dimostrare che la funzione $\dot{-}$, così definita: $x \dot{-} 0 = 0$ per $x \geq 0$, e $x \dot{-} \mathrm{succ}\,(y) = \mathrm{pd}\,(x \dot{-} y)$ per $x, y \geq 0$, è ricorsiva primitiva.

5) Giustificare intuitivamente le seguenti affermazioni.

   (a) Ogni insieme ricorsivo $A \subset \mathbb{N}$ è ricorsivamente enumerabile.

   (b) Il complementare $\mathbb{N} \setminus A$, di un insieme ricorsivo $A \subset \mathbb{N}$, è ricorsivamente enumerabile.

   (c) Se un insieme $A \subset \mathbb{N}$ ed il suo complementare $\mathbb{N} \setminus A$ sono ricorsivamente enumerabili, allora sono anche ricorsivi.

Logica 1
a.a. 2003/04
N. 7

Esercizi

1) Dimostrare che, per ogni $k \in \mathbb{N}$, la funzione $c_k\colon \mathbb{N} \to \mathbb{N}$ definita da $c_k(n) = k$ per ogni $n \in \mathbb{N}$ è ricorsiva primitiva.

2) Si dice che una funzione $f\colon \mathbb{N}^{k+1} \to \mathbb{N}$ è definita *per recursione* a partire dalle funzioni $g(x_1, \dots, x_k)$ e $h(x_1, \dots, x_k, y, z)$ se:

$$\begin{cases} f(x_1, \dots, x_k, 0) = g(x_1, \dots, x_k), \\ f(x_1, \dots, x_k, \mathrm{succ}(y)) = \\ \quad = h(x_1, \dots, x_k, y, f(x_1, \dots, x_k, y)). \end{cases} \tag{1}$$

Dimostrare che la moltiplicazione $f(x, y) = xy$ è ricorsiva primitiva. Precisamente, determinare due funzioni ricorsive primitive g e h in modo tale che f si possa definire con lo schema (1).

3) Dimostrare che il fattoriale $f(y) = y\,!$ è una funzione ricorsiva primitiva. Cosa sono g ed h in questo caso?

4) Dimostrare che l'elevamento a potenza $f(x, y) = x^y$ è una funzione ricorsiva primitiva. Determinare g e h come per gli esercizi precedenti.

5) Sia data una funzione ricorsiva $g(x, y)$. Ideare un algoritmo che calcoli la funzione f definita da $f(x) = \min\{\, y \mid g(x, y) = 0 \,\}$.

Logica 1
a.a. 2003/04
N. 6

Tema

Discutere la cardinalità dell'insieme di tutte le macchine di Turing e dedurne qualche conseguenza rilevante per la teoria della computabilità effettiva.

Logica 1
a.a. 2003/04
N. 5

Esercizi

1) Indicati con h e k due generici numeri interi, diversi fra loro, scrivere una funzione biunivoca $f_{h,k}\colon \mathbb{Z} \to \mathbb{Z} \setminus \{h, k\}$.

2) Scrivere una macchina di Turing che calcoli la funzione identica $\mathbf{1}_{\mathbb{N}}$.

3) Scrivere una macchina di Turing che calcoli la prima proiezione $\pi_1^2(x_1, x_2)$ a due argomenti.

4) Scrivere una macchina di Turing che calcoli la *seconda* proiezione $\pi_2^2(x_1, x_2)$ a due argomenti.

5) Scrivere una macchina di Turing che calcoli la funzione $\mathrm{succ}^2(x) = x + 2$.

6) Scrivere una macchina di Turing che calcoli la funzione $Z\colon \mathbb{N} \to \mathbb{N}$ definita come segue: $Z(x) = 0$ per ogni $x \in \mathbb{N}$.

7) Scrivere una macchina di Turing che calcoli la funzione $c_1\colon \mathbb{N} \to \mathbb{N}$ definita come segue: $c_1(x) = 1$ per ogni $x \in \mathbb{N}$.

8) Scrivere una macchina di Turing che, partendo dal nastro vuoto, scriva la successione $1{*}1{*}1{*}1{*}1{*}1{*}1\ldots$ procedendo verso destra (senza arrestarsi).

Logica 1
a.a. 2003/04
2-4-2004

Verifica

1) Per un generico $k \in \mathbb{Z}$, scrivere una funzione biunivoca $f_k\colon \mathbb{Z} \to \mathbb{Z} \setminus \{k\}$.

2) Indicato con A un insieme numerabile, studiare la cardinalità dell'insieme B definito come segue:

$$B = \{\, S \mid S \subset A,\ S \text{ è finito e non vuoto} \,\}.$$

Logica 1
a.a. 2003/04
N. 4

Esercizi

1) Indicato con S un sottoinsieme qualunque di $\mathbb{N}$, e con g una qualunque funzione da S verso $\mathbb{N}^{\mathbb{N}}$, dimostrare che g non è suriettiva.

2) Costruire una funzione $\bar{f}\colon \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$.

3) Studiare l'iniettività della funzione $\bar{f}$ costruita nell'esercizio precedente.

4) Indicata con f una funzione iniettiva da $\mathbb{N}^{\mathbb{N}}$ verso $\mathbb{N}$, e posto $S = f(\mathbb{N}^{\mathbb{N}})$, calcolare, per ogni $s \in S$, quante sono le successioni $a \in \mathbb{N}^{\mathbb{N}}$ che soddisfano l'equazione $f(a) = s$.

5) Siano f ed S come nell'esercizio precedente, e sia $g\colon S \to \mathbb{N}^{\mathbb{N}}$ la funzione che ad ogni $s \in S$ associa quell'unica successione $a \in \mathbb{N}^{\mathbb{N}}$ tale che $f(a) = s$. Studiare la suriettività di g.

6) Dimostrare che non esiste una funzione iniettiva da $\mathbb{N}^{\mathbb{N}}$ verso $\mathbb{N}$.

7) Studiare la cardinalità dell'insieme di quei numeri reali la cui rappresentazione decimale consta di un numero finito di cifre.

8) Studiare la cardinalità dell'insieme dei numeri periodici.

Logica 1
a.a. 2003/04
N. 3

Esercizi

1) Indicata con f una funzione biunivoca da $\mathbb{N}^2$ verso $\mathbb{N}$, si consideri la funzione $f_3: \mathbb{N}^3 \to \mathbb{N}$ definita come segue:
$$f_3(n_1, n_2, n_3) = f(n_1, f(n_2, n_3)).$$
Stabilire se f_3 è biunivoca.

2) Stabilire per quali valori di $k \in \mathbb{N}$ l'insieme $\mathbb{N}^{k+1}$ è numerabile.

3) Indicato con X un insieme non vuoto, si dice *parola* su X una qualunque sequenza ordinata di un numero finito di elementi di X (eventualmente anche uguali fra loro). L'insieme di tutte le parole su X si denota con X^*. Studiare la cardinalità dell'insieme S^* nell'ipotesi che S (sia non vuoto e) contenga un numero finito di elementi.

4) Studiare la cardinalità dell'insieme $\mathbb{N}^*$.

5) Stabilire se l'insieme $\mathbb{N}^{\mathbb{N}}$ è numerabile.

Logica 1
a.a. 2003/04
N. 2

Esercizi

1) Stabilire se esiste una funzione biunivoca $f\colon \mathbb{N} \to \{\, n \in \mathbb{N} \mid \exists k \in \mathbb{N} \quad (n = k^2) \,\}$. In caso affermativo, esibire una tale f; in caso negativo, dimostrarne la non-esistenza.

2) Dimostrare che l'unione di due insiemi numerabili è un insieme numerabile.

3) Definire una funzione biunivoca $f\colon \mathbb{N}^2 \to \mathbb{N}$.

4) Consideriamo una successione di insiemi A_k, e una successione di funzioni biunivoche $g_k\colon \mathbb{N} \to A_k$. Supponiamo, inoltre, che gli insiemi A_k siano a due a due disgiunti. Definiamo l'insieme A e la funzione $g\colon \mathbb{N}^2 \to A$ come segue:

$$A = \bigcup_{k=0}^{+\infty} A_k, \qquad g(k,n) = g_k(n).$$

Stabilire se g è biunivoca.

5) Dimostrare che *l'unione di un'infinità numerabile di insiemi numerabili e a due a due disgiunti è un insieme numerabile.* L'ipotesi *a due a due disgiunti* è essenziale?

6) Indicato con $\mathbf{2}$ l'insieme $\{\,0,1\,\}$, e con $\mathbf{2}^{\mathbb{N}}$ l'insieme $\{\, f \mid f\colon \mathbb{N} \to \mathbf{2} \,\}$, dimostrare che quest'ultimo non è numerabile.