**Università degli Studi di Udine**

Prova Scritta di *Fondamenti dell'Informatica II*

4 Dicembre 2002

1. Si enunci e dimostri il "Pumping Lemma" per i Linguaggi liberi dal contesto.

2. Si dia la definizione di "insieme semplice" e si spieghi, in breve, la ragione che ha motivato la ricerca di esistenza di tale tipo di insiemi.

3. Sia, per $i > 0$, $p_i$ l'$i$-esimo numero primo ($p_1 = 2, p_2 = 3, \ldots$) e $b(p_i)$ la sua rappresentazione come numero binario, senza 0 iniziali superflui. Si considerino, al variare di $i \in \mathbb{N}$, le seguenti famiglie di linguaggi:

$$\begin{array}{rcl} R_i & = & \left\{ x \in \{0\}^* \;\middle|\; \exists j \leq i.|x| = p_j \right\} \\ L_i & = & \left\{ x \in \{0,1\}^* \;\middle|\; \exists j \leq i.x = b(p_j) \right\} \end{array}$$

   (a) Si mostri che per ogni $i > 0$ $R_i$ e $L_i$ sono regolari.

   (b) Si descriva il DFA minimo per $L_4$.

   (c) Si dica se $R = \bigcup_{i>0} R_i$ è libero dal contesto, dimostrando formalmente la propria affermazione.

   (*) Si dica se $L = \bigcup_{i>0} L_i$ è libero dal contesto, dimostrando formalmente la propria affermazione.[1]

4. Sia $p$ la funzione definita nell'esercizio precedente. Si classifichi nella teoria matematica della ricorsione il seguente insieme:

$$\left\{ x \;\middle|\; \begin{array}{l} \varphi_{\lceil \log x \rceil}(p_1)) \downarrow \wedge \varphi_{\lceil \log x \rceil}(p_2) \downarrow \wedge \cdots \wedge \varphi_{\lceil \log x \rceil}(p_x) \downarrow \wedge \\ \varphi_{\lceil \log x \rceil}(p_x) = x \end{array} \right\}$$

5. Si consideri il seguente problema $ST(k)$: *Siano dati un grafo non diretto $G = \langle N, E \rangle$ e un intero $k \in \mathbb{N}$. Si vuol sapere se esiste uno SPANNING TREE per $G$ tale che nessun nodo dell'albero abbia grado maggiore di $k$.* In altri termini, se esiste $E' \subseteq E$ t.c.:

   - $|E'| = |N| - 1$,
   - il grafo $\langle N, E' \rangle$ è connesso, e
   - nessun nodo di $N$ è incluso in più di $k$ archi di $E'$.

   Si mostri che $ST(k)$ è NP-completo.

[1] Da farsi solo dopo aver risposto a tutte le altre domande!

# UNIVERSITÀ DEGLI STUDI DI UDINE

Prova Scritta di *Fondamenti dell'Informatica II*

8 Gennaio 2003

1. Si dimostri che i linguaggi liberi dal contesto sono chiusi per unione e concatenazione ma non per intersezione.

2. Si dimostri che ogni insieme produttivo ammette un sottoinsieme r.e. e infinito. Si accenni poi ad una conseguenza importante di questo risultato.

3. Si consideri, al variare di $i \in \mathbb{N}$, il seguente linguaggio:

$$L_i = \left\{ x \in \{0,1,+,=\}^* \;\middle|\; \begin{array}{l} |x| = i \wedge x \equiv u+v=z \wedge u,v,z \in \{0,1\}^+ \wedge \\ u,v,z \text{ sono rappr. binarie di numeri naturali,} \\ \text{senza prefissi di 0 inutili, e} \\ z \text{ rappr. la somma dei numeri rappr. da } u \text{ e } v. \end{array} \right\}$$

   (a) Si mostri che per ogni $i > 0$ $L_i$ è regolare.

   (b) Si descriva il DFA minimo per $L_5$.

   (c) Si dica se $L = \bigcup_{i>0} L_i$ è libero dal contesto, dimostrando formalmente la propria affermazione.

4. Sia, per $i > 0$, $p_i$ l'$i$-esimo numero primo ($p_1 = 2, p_2 = 3, \ldots$). Si dimostri che esiste un indice $u$ tale che $W_u = \{p_1, p_2, \ldots, p_{(u!)}\}$ e $E_u = \{1, 2, \ldots, u\}$.

5. Si dimostri la NP-completezza del *problema del cruciverba*, ovvero, dato un insieme di parole $p_1, \ldots, p_k$ in un dato alfabeto $A$ (N.B. potrebbe essere un alfabeto con più delle 26 lettere che usiamo di solito!) e uno schema di cruciverba $C$ (una matrice rettangolare di caselle bianche e nere) con le caselle nere già fissate, stabilire se esiste un modo per scrivere le $k$ parole nello schema, con le regole di incrocio standard dei cruciverba, che riempia esattamente tutte le caselle bianche.

   Può far comodo conoscere ed utilizzare il seguente problema, che è NP-completo:

   *Exact Cover by 3 sets*: Dato un insieme $U = \{1, \ldots, 3 \cdot m\}$, e dati $n$ insiemi $S_1, \ldots, S_n \subseteq U$, con $|S_1| = |S_2| = \cdots = |S_n| = 3$ stabilire se esistono $m$ insiemi (disgiunti) tra $S_1, \ldots, S_n$ tali che la loro unione sia $U$.

**UNIVERSITÀ DEGLI STUDI DI UDINE**
Prova Scritta di *Fondamenti dell'Informatica II*
4 Aprile 2003

1. Si enunci e dimostri il teorema di "equivalenza" tra DFA e NFA.

2. Si dimostri che esiste $x \in \mathbb{N}$ tale che

$$E_x = \{\lfloor \log x \rfloor + 1, \lfloor \log x \rfloor + 3, \lfloor \log x \rfloor + 5, \lfloor \log x \rfloor + 7, \ldots\}$$

   Si fornisca poi un "algoritmo" per il calcolo di tale $x$ (suggerimento: si ripercorra la dimostrazione del primo teorema di ricorsione).

3. Sia $L = \{a_1, a_1a_2, a_1a_2a_3, a_1a_2a_3a_4, \ldots\}$, dove $a_i$ è l'$i$-esima cifra dopo la virgola dell'espansione decimale di $\sqrt{2}$ ($a_1 = 4, a_2 = 1, a_3 = 4, a_4 = 2, \ldots$). Si dica se $L$ è ricorsivo, se $L$ è CF e se $L$ è regolare.

4. Si classifichi nella teoria matematica della ricorsione l'insieme

$$A = \left\{ \ x \,\middle|\, \varphi_x(3x+1) = x \ \right\}$$

5. Si consideri il problema CLIQUE:
   **Input:** Un grafo non diretto $G = \langle N, E \rangle$ e un intero $k$.
   **Problema:** Stabilire se esiste un sottografo di $G$ di $k$ elementi che sia un grafo completo.

   Si dimostri che CLIQUE è NP-completo.

**Università degli Studi di Udine**
Prova Scritta di *Fondamenti dell'Informatica II*
8 Luglio 2003

1. Si enunci e dimostri il teorema di Myhill-Nerode.
2. Si enunci e dimostri il primo teorema di ricorsione.
3. Per $i \geq 0$, si definisca la seguente famiglia di linguaggi:
$$\begin{cases} L_0 &= \left\{ 0^n \;\middle|\; n \text{ è multiplo di } 3 \right\} \\ L_{i+1} &= L_i \cdot \{0\} \end{cases}$$
   (a) $L_i$ è regolare? Se sí, si descriva il DFA minimo per $L_{10}$ e la grammatica in forma normale di Chomsky che lo genera.
   (b) $\bigcup_{i\geq 0} L_i$ è regolare? Si motivi la risposta (affermativa o negativa).
4. Si collochi nella teoria matematica della ricorsione il seguente insieme
$$A = \left\{ x \;\middle|\; \exists u \exists v\, (x = \langle u, v\rangle \wedge \varphi_{uv}(u+v) = 0) \right\}$$
Si assuma, al solito, che $\langle \cdot, \cdot \rangle$ sia la funzione biiettiva di codifica delle coppie di naturali in naturali.
5. Si consideri il problema che chiamiamo ZSAT:
**Input:** Una congiunzione di clausole $C_1 \wedge \cdots \wedge C_n$ ove ogni $C_i$ è una disgiunzione di ESATTAMENTE $i$ letterali (non necessariamente diversi tra loro).
**Problema:** Stabilire se esiste un assegnamento di verità per le variabili che compaiono nella formula che rende l'istanza vera.

   Si dimostri che ZSAT è NP-completo.

**UNIVERSITÀ DEGLI STUDI DI UDINE**

Prova Scritta di *Fondamenti dell'Informatica II*

22 Luglio 2003

1. Si dia la definizione di insieme semplice e si dimostri l'esistenza di un insieme semplice.

2. Si dimostri che

$$L = \left\{ \ x \in \{0,1\}^* \ \middle| \ \# \text{ di } 0 \text{ in } x = \# \text{ di } 1 \text{ in } x \ \right\}$$

   non è regolare. Si mostri poi che è libero dal contesto e si fornisca una grammatica generatrice in forma normale di Greibach.

3. Si collochi nella teoria matematica della ricorsione il seguente insieme:

$$C = \left\{ \ x \,\middle|\, \forall i \leq x \, (\varphi_x(i) = i!) \ \right\}$$

4. Si mostri che esiste $\nu \in \mathbb{N}$ tale che:

$$\varphi_\nu(y) = \begin{cases} 2^y & y \leq \nu \\ 5 & y > \nu \end{cases}$$

5. Si consideri il problema della fila tra detenuti (FTD): vi sono $n$ detenuti ed è nota una relazione $x$ *odia* $y$ tra coppie di detenuti. Il problema è: esiste un modo per mettere in fila gli $n$ detenuti in modo tale che non vi sia mai un detenuto con uno che lo odia alle spalle?

   Si mostri che il problema FTD è NP-completo.

# Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

9 Settembre 2003

1. Si dimostri che $K$ è completo, dando le definizioni necessarie. Si enunci e facoltativamente dimostri (fatelo alla fine eventualmente) il teorema di Myhill.

2. Si consideri la seguente famiglia di linguaggi:

$$\begin{cases} L_0 &= \{\varepsilon\} \\ L_{i+1} &= \left\{ ua^xb^y \;\middle|\; u \in L_i \land x > i \land y > i \right\} \end{cases}$$

   (a) Si mostri che, fissato $i$, $L_i$ è regolare

   (b) Si mostri che $L = \bigcup_{i\in\mathbb{N}} L_i$ non è libero dal contesto.

3. Si dimostri che esiste $x \in \mathbb{N}$ tale che $E_x = \{x, x^2, x^3, x^4, \dots\}$.

4. Sia $g$ una data funzione ricorsiva totale e iniettiva. Si collochi nella gerarchia della ricorsione l'insieme $I = \left\{ g(x) \;\middle|\; \varphi_x(0)\downarrow \land \varphi_x(g(x))\downarrow \right\}$. E se manca l'ipotesi di iniettività?

5. Si definisca il problema 2SAT. Lo si collochi nella corretta classe di complessità temporale.

**Università degli Studi di Udine**

Prova Scritta di *Fondamenti dell'Informatica II*

8 Gennaio 2004

1. Si dia la definizione di insieme produttivo; si dimostri dunque che ogni insieme produttivo contiene un insieme ricorsivamente enumerabile infinito.

2. Si definisca un problema NP-completo di interesse *pratico* (si motivi l'interesse pratico del problema proposto) e si dimostri la sua NP-completezza.

3. Si consideri il linguaggio:

$$L = \left\{ 1^{4m}2^{3n}3^{2n}4^{m} \;\middle|\; m \geq 0, n \geq 0 \right\}$$

   (a) Si mostri che $L$ non è regolare

   (b) Si mostri che $L$ è libero dal contesto e si fornisca una grammatica in forma normale di Greibach che lo genera.

   (c) Si mostri che $L' = \left\{ 1^{4n}2^{3m}3^{2n}4^{n} \;\middle|\; m \geq 0, n \geq 0 \right\}$ non è libero dal contesto.

4. Si dimostri che esiste $x \in \mathbb{N}$ tale che $W_x = \mathbb{N} \setminus \{x\}$.

5. Si collochi nella gerarchia della ricorsione (ovvero si dica se è ricorsivo, ricorsivamente enumerabile, creativo, produttivo, completo, semplice, ecc.) l'insieme:

$$A = \left\{ x \;\middle|\; \exists y \in \mathbb{N} \, (\varphi_y(x) \downarrow \wedge \varphi_x(y) \downarrow \wedge \varphi_x(y) \leq \varphi_y(x)) \right\} .$$

UNIVERSITÀ DEGLI STUDI DI UDINE
Prova Scritta di *Fondamenti dell'Informatica II*
16 Marzo 2004

1. Si enunci e dimostri il teorema di "equivalenza" tra DFA e NFA.
2. Si dimostri la proprietà: *Se $A$ è produttivo allora $A$ non è ricorsivamente enumerabile.*
3. Sia data la seguente famiglia di linguaggi, al variare di $i \in \mathbb{N}$:

   $$L_i \quad = \quad \left\{ \, x \in \{0,1\}^* \;\middle|\; \#(0,x) = 2^i \, \right\}$$

   dove $\#(0,x)$ restituisce il numero di occorrenze del simbolo 0 nella stringa $x$.

   (a) Si dimostri che per ogni $i$ $L_i$ è un linguaggio regolare.
   (b) Si fornisca una grammatica in forma normale di Chomsky per $L_2$.
   (c) Si mostri che $L = \bigcup_{i \geq 0} L_i$ non è libero dal contesto.
4. Si *dimostri* nel modo più semplice possibile che

   $$A = \left\{ \, x \;\middle|\; W_x = \{0,1,2,3\} \, \right\}$$

   non è ricorsivo. Cosa ci permette di dire su $A$ il Teorema di Rice-Shapiro?
5. Si collochi nella gerarchia della ricorsione l'insieme:

   $$B = \left\{ \, x \;\middle|\; \exists y \exists z \, (x = \langle y, z \rangle \wedge \varphi_{y \, div \, 2}(y^2+1) = \lceil \sqrt{y} \rceil \wedge \varphi_z(0) = 1) \, \right\}$$

6. Si mostri che il complementare dell'insieme $A$ dell'esercizio (4) è produttivo e si evidenzi la sua funzione produttiva (suggerimento: si segua la dimostrazione della proprietà *se $X$ è produttivo e $X \preceq Y$ allora $Y$ è produttivo.*)

**UNIVERSITÀ DEGLI STUDI DI UDINE**

Prova Scritta di *Fondamenti dell'Informatica II*

7 Aprile 2004

1. Si enunci e dimostri il primo teorema di ricorsione.

2. Si enunci il problema dell'insieme indipendente e si dimostri la sua NP-completezza.

3. Si dica se valgono o meno (e perché) le riduzioni $K \preceq \bar{K}$ o $\bar{K} \preceq K$.

4. Si studi il linguaggio:

$$A = \left\{ x2y2z \,\middle|\, x, y, z \in \{0,1\}^*, \#(0,x) + \#(1,y) = \#(0,z) \right\}$$

   dove $\#(s,v)$ restituisce il numero di occorrenze del simbolo $s$ nella stringa $v$. In particolare si

   (a) dimostri se è o meno regolare. Nel caso lo sia, si fornisca il DFA minimo che lo accetta.

   (b) dimostri se è o meno libero dal contesto. Nel caso lo sia, si fornisca una grammatica in normale di Chomsky che lo genera.

5. Si considerino gli insiemi

$$B = \left\{ x \,\middle|\, \varphi_x(0) = \varphi_x(1) = \cdots = \varphi_x(100) \right\}$$

$$C = \left\{ x \,\middle|\, \sqrt{\varphi_x(2x+1)} = x \right\}$$

   Per ciascuno dei due si dica se è o meno:

   (a) estensionale

   (b) ricorsivo

   (c) r.e.

   (d) produttivo

   (e) completo

   Si dimostri dunque che $B \preceq C$.

6. Si studino l'insieme $D = \left\{ x \,\middle|\, E_x = \{x\} \right\}$

**UNIVERSITÀ DEGLI STUDI DI UDINE**
Prova Scritta di *Fondamenti dell'Informatica II*
13 Luglio 2004

1. Si enunci il Teorema di Myhill-Nerode e se ne dimostri almeno uno dei tre punti.

2. Si mostri che se $\bar{K} \preceq A$ allora $A$ è produttivo.

3. Si studi il linguaggio:

$$A \;=\; \left\{\ a_0a_1\cdots a_n \,\middle|\, n > 0, (\forall i \in \{0,\ldots,n\})(a_i \in \{0,1\}), (\textstyle\sum_{i=0}^{n} a_i 2^i) \ \mathbf{mod}\ 12 = 0\ \right\}$$

   Qualora sia regolare si fornisca l'automa minimo che lo accetta.

4. Si mostri che esiste $x$ tale che $E_x = \{x, x+2, x+4, x+6, \ldots\}$.

5. Si studi l'insieme

$$C = \left\{\ x \,\middle|\, (\forall y \leq 1000)(\varphi_x(y)\downarrow) \wedge (\forall y < 1000)(\varphi_x(y) \leq \varphi_x(y+1)) \wedge \varphi_x(2500) = 3\ \right\}.$$

   In particolare si dica, motivando la propria affermazione, se sia o meno estensionale, ricorsivo, r.e., semplice o completo.

6. Si consideri l'insieme

$$B \;=\; \left\{\ x \,\middle|\, W_x = \{2^0, 2^1, 2^2, 2^3, 2^4, \ldots\}\ \right\}$$

   Si mostri che sia $B$ che $\bar{B}$ sono produttivi.

**Università degli Studi di Udine**
Prova Scritta di *Fondamenti dell'Informatica II*
2 Settembre 2004

1. Sia $G$ una grammatica libera dal contesto. Si dimostri che il problema $x \in L(G)$? è decidibile.

2. Si dia la definizione di insieme produttivo. Se ne fornisca un esempio e si dimostri dunque che ogni insieme produttivo contiene un insieme r.e. infinito.

3. Sia $\Sigma = \{0, 1, 2\}$ un alfabeto e $\preceq \subseteq \Sigma \times \Sigma$ la relazione definita estensionalmente nel modo seguente: $0 \preceq 0, 0 \preceq 1, 0 \preceq 2, 1 \preceq 1, 1 \preceq 2, 2 \preceq 2, 2 \preceq 0$. Si considerino i seguenti due linguaggi:

$$\begin{array}{rcl} A & = & \left\{ a_0 a_1 \ldots a_n \in \Sigma^* \,\middle|\, n > 0 \wedge (\forall i \in \{0, \ldots, n-1\})(a_i \preceq a_{i+1}) \wedge \sum_{i=0}^{n} a_i \equiv_3 0 \right\} \\ B & = & \left\{ a_0 a_1 \ldots a_n \in A \,\middle|\, (\exists k \in \mathbb{N})(n = 2^k) \right\} \end{array}$$

Si dimostri in modo rigoroso che $A$ è regolare. Si dimostri inoltre se $B$ sia o meno libero dal contesto.

4. Si studino l'insieme $C$ ed il suo complementare:

$$C = \left\{ x \,\middle|\, (\forall y \in \mathbb{N})(\varphi_x(y+1) \downarrow \wedge \varphi_x(y) \downarrow \wedge \varphi_x(y+1) > \varphi_x(y)) \right\}$$

5. Sia $\eta$ una funzione calcolabile che associa ad ogni programma `while` scritto in sintassi concreta un diverso numero naturale. Si assuma di saper calcolare anche l'inversa di $\eta$ (dato un numero, restituire un programma `while` oppure la costante $\perp$). Si denoti con $\mathcal{W}$ l'insieme dei programmi while e con $|P|$ il numero di simboli (la lunghezza) di un programma while, tolti gli spazi ridondanti e i simboli di a capo. Si studi l'insieme:

$$D = \left\{ x \,\middle|\, (\exists P \in \mathcal{W})(x = \eta(P)) \wedge (\forall Q \in \mathcal{W})(|Q| \leq |P| \longrightarrow P(Q) \downarrow) \right\}$$

Si definisca (anche a parole) una possibile funzione $\eta$ con le caratteristiche sopra.

6. In che classe di complessità si collocano gli insiemi $A$ e $B$ dell'esercizio 3? Se avanza tempo, si costruisca il DFA minimo per l'insieme $A$.

Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

22 Settembre 2004

1. Si diano con precisione le definizione di riduzione tra classi di complessità e di completezza. Si definiscano dunque la classe $P$ ed un problema $P$-completo.

2. Si definisca l'insieme semplice, si mostri che ogni insieme semplice non è né ricorsivo né completo e si descriva un insieme semplice noto.

3. Si dimostri che
$$L \;=\; \{\; 0^m 1^n 0^p 1^{2p} \mid m, n, p \in \mathbb{N} \;\}$$
è libero dal contesto (si individui una $G$ tale che $L = L(G)$ e si dimostri che $L \subseteq L(G)$ e $L(G) \subseteq L$).

4. Si dimostri che $L$ non è regolare.

5. Si dimostri che esiste $x$ tale che
$$E_x = \{0, 1, 2, 3, \ldots, a_0 \cdot a_1 \cdot a_2 \cdot \cdots \cdot a_x\}$$
ove, per $i = 0, \ldots, x$, $a_x$ è l'a $x$-esima cifra dell'espansione decimale di $e^x$ (prima e dopo la virgola).

6. Sia $\eta$ una funzione calcolabile che associa ad ogni programma `while` scritto in sintassi concreta un diverso numero naturale. Si assuma di saper calcolare anche l'inversa di $\eta$ (dato un numero, restituire un programma `while` oppure la costante $\perp$). Si denoteranno con $\mathcal{W}$ l'insieme dei programmi `while`, e con $\underline{n}$ la rappresentazione mediante numerali del numero naturale $n$. Si studi dunque l'insieme:
$$A = \{\; x \mid (\exists P \in \mathcal{W})(x = \eta(P)) \wedge (\forall y < x)([\![P]\!](\underline{y}) = \underline{2^y}) \;\}$$

# Università degli Studi di Udine

## Prova Scritta di *Fondamenti dell'Informatica II*

## 4 Gennaio 2005

1. Si definisca il concetto di insieme semplice e si mostri che un insieme semplice non è né ricorsivo né completo.
2. Si dimostri che $L = \{a^n b^n c^n : n \in \mathbb{N}, n > 2\}$ non è libero dal contesto.
3. Si consideri il linguaggio MINI definito come segue:
   - Vi è un insieme $\{X_0, X_1, X_2, X_3, \ldots\}$ di variabili atte a contenere numeri naturali e inizializzate per default a 0.
   - Le istruzioni di MINI sono definite dalla seguente grammatica:

$$\begin{array}{rcl} \langle istruz \rangle & ::= & \langle assegnamento \rangle | \langle for \rangle | \langle istruz \rangle ; \langle istruz \rangle \\ \langle assegnamento \rangle & ::= & \langle var \rangle = 0 | \langle var \rangle = \langle var \rangle | \langle var \rangle = \langle var \rangle + 1 \\ \langle for \rangle & ::= & \mathbf{for} \langle var \rangle = \langle var \rangle \mathbf{to} \langle var \rangle \mathbf{do} \langle istruz \rangle \mathbf{endfor} \end{array}$$

   - Un programma MINI è semplicemente una istruzione.
   - La semantica delle varie istruzioni è quella intuitiva. Ad esempio, se $X_2 = 5$ e $X_3 = 9$ allora $\mathbf{for} X_1 = X_2 \mathbf{to} X_3 \mathbf{do} X_2 = X_2 + 1 \mathbf{endfor}$ esegue 5 volte l'istruzione $X_2 = X_2 + 1$.
   - Un programma $P$ accetta in input $n$ valori che servono ad inizializzare le variabili $X_1, \ldots, X_n$.
   - Il risultato del programma è il valore finale della variabile $X_0$.

   Si mostri che MINI è in grado di calcolare tutte le funzioni primitive ricorsive.
4. Sia $f : \mathbb{N} \times \mathbb{N} \longrightarrow \mathbb{N}$ una funzione ricorsiva totale e biiettiva. Si studino i due insiemi:

$$\begin{array}{rcl} A & = & \{f(x, y) : W_x \cap W_y = \emptyset\} \\ B & = & \{f(x, y) : W_x \subseteq W_y\} \end{array}$$

5. Si consideri un sentiero con un certo numero di sbarramenti (diciamo $\ell$) costituiti ciascuno da 3 cancelli. Ogni cancello è etichettato con un numero naturale tra 1 e $p$, con $p$ numero pari. Vi possono essere più cancelli etichettati con lo stesso numero. Alcune etichette possono non essere usate.

   Una *configurazione* è un insieme di etichette $S \subseteq \{1, \ldots, p\}$ tale che per ogni numero dispari $x \in \{1, \ldots, p\}$, esattamente uno tra $x$ e $x + 1$ sta in $S$.

   Data una configurazione $S$, tutti i cancelli etichettati con elementi di $S$ sono aperti, gli altri sono chiusi. Nell'esempio seguente, $\ell = 2$, $p = 4$, e $S = \{1, 4\}$ e i cancelli etichettati con 1 sono aperti, lasciando un cammino nel sentiero.

$$\begin{array}{ccc} & 1 & \boxed{2} \\ \Rightarrow & \boxed{3} & 1 \\ & 1 & \boxed{2} \end{array}$$

   Si dimostri che il problema di stabilire se esiste una configurazione che permette un cammino nel sentiero è NP-completo.

**Università degli Studi di Udine**
Prova Scritta di *Fondamenti dell'Informatica II*
15 Marzo 2005

1. Si dimostri che, data $G$ grammatica libera dal contesto, il problema $x \in L(G)$ è decidibile.

2. Si definiscano le nozioni di (r.e.) completezza e di creatività. Si dimostri dunque che i due concetti sono equivalenti.

3. Si considerino i linguaggi:

$$\begin{array}{rcl} A & = & \{0^x 1^y \,:\, x \leq y \wedge x < 4\} \\ B & = & \{uu^r \,:\, u \in A\} \\ C & = & \{uu^r \,:\, u \in B\} \end{array}$$

ove $u^r$ è definito induttivamente come segue: $\varepsilon^r = \varepsilon$ e $(av)^r = (v^r)a$.

   (a) Si mostri che $A$ è regolare e si fornisca il DFA minimo.

   (b) Si mostri che $B$ è libero dal contesto. $B$ è regolare?

   (c) Si mostri se $C$ è o meno libero dal contesto.

4. Si studino gli insiemi

$$\begin{array}{rcl} D & = & \{\langle x, y\rangle \,:\, (\forall z \leq y)(\varphi_x(z) = 2y) \wedge \varphi_x(2y) = y^2 - 1\} \\ E & = & \{\langle x, y\rangle \,:\, (\forall z \leq y)(\varphi_x(z) \uparrow) \wedge (\forall z > y)(\varphi_x(z) = x)\} \end{array}$$

5. Si consideri un sentiero con un certo numero di sbarramenti (diciamo $\ell$) costituiti ciascuno da 3 cancelli. Ogni cancello è etichettato con un numero naturale tra 1 e $p$. Vi possono essere più cancelli etichettati con lo stesso numero. Alcune etichette possono non essere usate.

   Una *configurazione* è un insieme di etichette $S \subseteq \{1, \ldots, p\}$ tale che per ogni $x \in S$, se $x$ è dispari, allora $x + 1 \notin S$, se $x$ è pari, allora $x - 1 \notin S$ e per ogni numero dispari $x \in \{1, \ldots, p\}$, almeno uno tra $x$ e $x + 1$ sta in $S$.

   Data una configurazione $S$, tutti i cancelli etichettati con elementi di $S$ sono aperti, gli altri sono chiusi. Nell'esempio seguente, $\ell = 2$, $p = 4$, e $S = \{1, 4\}$. I cancelli etichettati con 1 sono aperti, lasciando un cammino nel sentiero.

$$\begin{array}{ccc} & 1 & \boxed{2} \\ \Rightarrow & \boxed{3} & 1 \\ & 1 & \boxed{2} \end{array}$$

   Si dimostri che il problema di stabilire se esiste una configurazione che permette un cammino nel sentiero è NP-completo.

# Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

6 Aprile 2005

1. Si dimostri che: data una grammatica $G$ libera dal contesto tale che $L(G) \neq \emptyset$, esiste una grammatica $G'$ equivalente ma priva di simboli inutili.

2. Si dia la definizione di insieme produttivo e si dimostri che tale insieme non può essere né ricorsivo né r.e.

3. Si collochino nella gerarchia di Chomsky i due linguaggi seguenti:

$$\begin{array}{rcl} A & = & \{x \in \{0,1\}^* : (\sharp(0,x) > \sharp(1,x)) \longrightarrow (\sharp(0,x) \text{ è pari})\} \\ B & = & \{xxx : x \in A \text{ e } |x| = 2^n \text{ per qualche } n \in \mathbb{N}\} \end{array}$$

   ove $\sharp(a,x)$ è il numero di occorrenze del simbolo $a$ nella stringa $x$.

4. Si dimostri che esistono $x \in \mathbb{N}$ e $y \in \mathbb{N}$ tali che

$$W_{x+y} = \{x + y\}$$

5. Si studino dal punto di vista della teoria della ricorsione i seguenti insiemi:

$$\begin{array}{rcl} C & = & \{x : (\forall y \in \mathbb{N})(y < x \longrightarrow W_y \neq \emptyset)\} \\ D & = & \{\eta(P) : P \text{ programma While e } [\![P]\!]^W(P) = \underline{|P|^2}\} \\ E & = & \{\langle x, y, z\rangle : W_x = W_y \wedge W_z \subseteq W_x\} \end{array}$$

   ove $\eta$ è una funzione biiettiva e calcolabile di enumerazione dei programmi While, $\underline{n}$ il numerale associato al numero $n$ e $|P|$ la lunghezza (numero caratteri) di un programma While $P$.

6. Il problema $k$-coloring è il problema di stabilire se i nodi di un grafo $G = \langle N, E\rangle$ possano essere colorati con $k$-colori in modo tale che nodi adiacenti abbiano colore diverso. Ovviamente il problema ha sempre soluzione positiva per $k \geq |N|$. Si mostri che per $k \geq 3$ il problema è in NP e che per $k = 2$ il problema è in P.

| |
|---|
| **UNIVERSITÀ DEGLI STUDI DI UDINE**<br>Prova Scritta di *Fondamenti dell'Informatica II*<br>19 Luglio 2005 |

1. Si mostri che ogni funzione Turing calcolabile è definibile con il formalismo delle funzioni ricorsive (parziali).

2. Si mostri che, se $A$ è r.e. e $A \cap B$ è produttivo, allora $B$ è produttivo.

3. Si dimostri se $A = \left\{ 0^a 1^b 0^a 1^b \,\middle|\, a \in \mathbb{N}, b \in \mathbb{N} \right\}$ è o meno CF.

4. Si mostri che esiste $x \in \mathbb{N}$ per cui $E_x = \left\{ y \in \mathbb{N} \,\middle|\, y \text{ è pari e } y \leq x \right\}$

5. Sia $r \in \mathbb{N}$ fissato. Si studi l'insieme

$$B = \left\{ \langle x, y, z \rangle \in \mathbb{N} \,\middle|\, W_x \subseteq W_y \wedge E_x \subseteq E_z \right\}$$

6. Si consideri il seguente gioco. Ci sono $n$ caselle, ciascuna delle quali può contenere un intero dell'insieme $S = \{-m, -m+1, \ldots, -1, 1, \ldots, m-1, m\}$. Se una casella contiene $x \in S$, nessun'altra può contenere $-x$. Per ognuna delle caselle $i$ è fornito inoltre un insieme $S_i \subseteq S$ di possibili valori per tale casella.

   Si collochi nella opportuna classe di complessità il problema di stabilire se esiste un modo per riempire tutte le caselle che rispetti le semplici regole sopra.

   Ad esempio, se $n = 4, m = 2$ e $S_1 = \{1, 2\}, S_2 = \{-1, -2\}, S_3 = \{1, -2\}, S_4 = \{-1, 2\}$ non ammette soluzione, mentre con $n = 3$, togliendo $S_4$ vi sarebbe la soluzione $\boxed{1}\boxed{-2}\boxed{1}$.

# Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

6 Settembre 2005

1. Si enunci e dimostri il Teorema di Myhill.
2. Assumendo i teoremi di Cook, si dimostri dettagliatamente che 3SAT è NP completo.
3. Si consideri la seguente gerarchia di linguaggi, al variare di $i \in \mathbb{N}$:

$$\begin{array}{rcl} L_0 & = & \{0,1\}^* \\ L_i & = & L_{i-1} \setminus \{0^i 1^i\} \quad \text{con } i > 0 \end{array}$$

   (a) Si mostri che, dato $i \in \mathbb{N}$, $L_i$ è regolare e si fornisca l'automa minimo per $L_3$.

   (b) Si dica (dimostrandolo) se $L = \bigcap_{i \in \mathbb{N}} L_i$ è o meno regolare.

4. Si dimostri che $A = \left\{ x \,\middle|\, (\exists y > 0)(\varphi_x(x \cdot y)\downarrow) \right\}$ è creativo.
5. Sia $f : \mathbb{N}^3 \longrightarrow \mathbb{N}$ una funzione ricorsiva totale e biiettiva. Si studi l'insieme:

$$B \quad = \quad \left\{ f(x,y,z) \,\middle|\, W_x \cap W_y = \{0\} \wedge W_x \cap W_z = \{1\} \right\}$$

6. Si dica (dimostrandolo) se $L$ dell'esercizio (3b) è o meno libero dal contesto.

**UNIVERSITÀ DEGLI STUDI DI UDINE**
Prova Scritta di *Fondamenti dell'Informatica II*
20 Settembre 2005

1. Si mostri che esiste una funzione $f : \mathbb{N} \longrightarrow \mathbb{N}$ non Turing-calcolabile.
2. Si dimostrino o si refutino i seguenti enunciati:
   (a) Se $A$ è r.e. e $A \cap B$ è produttivo, allora $B$ è produttivo.
   (b) Se $A$ è ricorsivo e $B$ è produttivo, allora $A \cap B$ è produttivo.
3. Si dimostri che $\texttt{PAL} = \left\{ x \in \{0,1\}^* \;\middle|\; x \text{ è palindroma } \right\}$:
   (a) non è regolare;
   (b) è CF;
   (c) sta in NP. E' NP-completo?
4. Si studi l'insieme
$$A = \left\{ x \;\middle|\; (\forall y \leq x)(\varphi_y \text{ è totale}) \right\}$$
5. Si confrontino, con le relazioni $\subseteq$ e $\leq$, gli insiemi:
$$\begin{array}{rcl} B & = & \left\{ x \;\middle|\; \varphi_x(2x) \downarrow \right\} \\ C & = & \left\{ x \;\middle|\; \varphi_x \neq \lambda x. \uparrow \right\} \end{array}$$
6. Si studi l'insieme:
$$D = \left\{ \langle x, y, z \rangle \;\middle|\; (\forall u \leq z)(\varphi_x(u) \downarrow \rightarrow (\varphi_y(u) \downarrow \wedge \varphi_x(u) \leq \varphi_y(u))) \right\}$$

# Università degli Studi di Udine
Prova Scritta di *Fondamenti dell'Informatica II*
22 Marzo 2006

1. Si dimostri che, date $G_1$ e $G_2$ grammatiche libere dal contesto, il problema $x \in L(G_1) \cap L(G_2)$ è decidibile.

2. Si definiscano le nozioni di (r.e.) completezza e di creatività. Si dimostri dunque che $K_2$ è r.e. completo (senza utilizzare altri insiemi r.e. completi nella dimostrazione).

3. Dato $X = \{00, 01, 10\}$, siano $A$ e $B$ i seguenti linguaggi.

$$\begin{array}{rcl} A & = & \{v_1 v_2 \cdots v_n \,:\, n > 0, v_i \in X\} \\ B & = & \{w \in A \,:\, w \text{ è palindroma}\} \end{array}$$

   (a) Si mostri che $A$ è regolare e si fornisca il DFA minimo.

   (b) Si mostri che $B$ è libero dal contesto (e si fornisca una grammatica in forma normale di Chomsky che lo genera) ma **non** regolare.

4. Dati gli insiemi

$$\begin{array}{rcl} C & = & \{x \,:\, W_x \subseteq \{\text{numeri pari}\}\} \\ D & = & \{x \,:\, E_x \subseteq \{\text{numeri dispari}\}\} \end{array}$$

   Si mostri che $C \preceq D$.

5. Si fissi $\ell \in \mathbb{N}, \ell > 0$. Si studi (est., r.e., ric., compl., prod, etc.) l'insieme

$$E \quad = \quad \{x \,:\, (\forall y \leq \ell)(\varphi_x(y) = x + y)\}$$

6. Abbiamo a disposizione $n$ calciatori nervosi e vogliamo organizzare un torneo di calcio a $k > 2$ squadre. E' noto che tra alcune coppie di calciatori vi è un odio consolidato bidirezionale. Due calciatori che si odiano non possono stare in squadra assieme (so che state pensando che in squadre avverse tenteranno di menarsi in campo, ma questo sarà un problema dell'arbitro). Si mostri che stabilire se esiste un modo per fissare le $k$ squadre col vincolo di non odio tra compagni di squadra (non preoccupatevi di raggiungere un numero minimo di giocatori per squadra) è un problema NP-completo.

Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

04 Aprile 2006

1. Si enunci e dimostri il Teorema di Myhill-Nerode.

2. Si definisca la nozione di insieme semplice e si dimostri che un insieme semplice non è nè ricorsivo nè r.e. completo.

3. Sia $\mathsf{fibo} : \mathbb{N} \longrightarrow \mathbb{N}$ la funzione di fibonacci. Sia $n \in \mathbb{N}$. Definiamo

$$\begin{array}{rcl} A_n & = & \left\{ 0^n 1 0^m \mid m \geq \mathsf{fibo}(n) \right\} \\ B & = & \bigcup_{n \geq 0} A_n \end{array}$$

   (a) Si mostri che, dato $n$, $A_n$ è regolare e si fornisca il DFA minimo nel caso di $n = 3$.

   (b) Si studi $B$. Qualora sia libero dal contesto, si fornisca una grammatica in forma normale di Chomsky che lo genera.

4. Dati gli insiemi

$$\begin{array}{rcl} C & = & \left\{ x \mid W_x = \{0\} \right\} \\ D & = & \left\{ x \mid E_x = \{0,1\} \right\} \end{array}$$

   Si mostri che $C \preceq D$.

5. Sia data $\mathsf{zip} : \{0,1\}^* \longrightarrow \{0,1\}^*$ una funzione ricorsiva che comprime files binari (se vi fa comodo, pensate ad un programma di compressione files che usate nel vostro PC). Si assuma esista una funzione $\eta$ di enumerazione (biettiva) dei programmi ad un argomento scritti in **C** (se $P$ è un programma **C** ad un argomento, dunque, $\eta(P) \in \mathbb{N}$). Si collochi nella gerarchia della calcolabilità l'insieme:

$$E = \left\{ \eta(P) \mid P \text{ prg C ad un argomento e } (\forall y \in \{0,1\}^*)(P(y) = \mathsf{zip}(y)) \right\}$$

6. Si consideri un formalismo per il calcolo schematicamente definito come segue:

   **espressioni di dominio** Ad ogni variabile che vogliamo usare dobbiamo preventivamente fornire un dominio finito, ovvero un insieme finito di valori (scelti in $\mathbb{Z}$) ammissibili (ad esempio: $X \in \{0,1,2\}, Y \in \{1,2\}$)

   **vincoli** Possiamo scrivere formule del tipo $\ell = r$ (equazioni), $\ell \leq r$ e $\ell \neq r$ (disequazioni) ove i vari termini $\ell$ e $r$ sono variabili o numeri interi o si ottengono sommando o sottraendo variabili e/o numeri interi (ad esempio: $X + Y \leq 5, 1 \leq X - Y$).

   **CSP** Un CSP è costituito da un insieme di vincoli su delle variabili e dalle espressioni di dominio per **tutte** le variabili presenti.

   **soluzione** Una soluzione di un CSP è un assegnamento per le variabili presenti usando valori dei rispettivi domini che soddisfa tutti i vincoli (nell'esempio sopra, ad esempio: $X = 2, Y = 1$ soddisfa $2 + 1 \leq 5$ e $1 \leq 2 - 1$).

   Si dimostri che il problema di stabilire l'esistenza di una soluzione per un CSP è un problema NP-completo.

# Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

20 Luglio 2006

1. Si dia la definizione di insieme produttivo e si dimostri che tale insieme non può essere né ricorsivo né r.e.

2. Si dimostri che i linguaggi liberi dal contesto sono chiusi per unione e concatenazione ma non per intersezione.

3. Data una funzione $f : \mathbb{N} \longrightarrow \mathbb{N}$ denotiamo con $M_f$ un algoritmo su stringhe tale che per ogni $x \in \{0, 1\}^*$, se $x$ ha la forma $0^n 1^{f(n)}$, allora $M_f(x) \downarrow \texttt{yes}$, altrimenti $M_f(x) \downarrow \texttt{no}$.

   Si studi se $M_f$ possa essere implementata da un DFA o da un automa a pila non deterministico o da nessuno di questi automi, nei seguenti casi:

   (a) $(\forall x \in \mathbb{N}) f_1(x) = 5$
   (b) $(\forall x \in \mathbb{N}) f_2(x) = x$
   (c) $(\forall x \in \mathbb{N}) f_3(x) = \lceil \log_2 x \rceil$

4. In quali classi di complessità ricadono gli insiemi di stringhe accettate da $M_{f_1}, M_{f_2}, M_{f_3}$?

5. Sia $\eta$ una funzione biiettiva (calcolabile, come la sua inversa) di enumerazione dei programmi C ad un argomento (che assumiamo essere un intero senza limitazioni di lunghezza). Con $|P|$ denotiamo la lunghezza in bytes di un programma $P$. Con $W_P = \{ \ x \mid \ [\![P]\!](x) \downarrow \ \}$. Si studino i seguenti insiemi

   (a) $A = \{ \ \eta(P) \mid \ (\forall x \leq |P|)[\![P]\!](x) = x \ \}$
   (b) $B = \{ \ \eta(P) \mid \ [\![P]\!](0) \uparrow \ \}$
   (c) $C = \{ \ \langle \eta(P), \eta(Q) \rangle \mid \emptyset \subset W_P \subset W_Q \subset \mathbb{N} \ \}$

**UNIVERSITÀ DEGLI STUDI DI UDINE**
Prova Scritta di *Fondamenti dell'Informatica II*
5 settembre 2006

1. Si ripeta la definizione dell'automa basato sulla relazione $R_L$ come nel Teorema di Myhill-Nerode e se ne dimostri dunque l'unicità a meno di isomorfismi.

2. Si definisca con precisione la nozione di NP completezza. Assumendo noti i Teoremi di Cook, si dimostri che 3SAT è NP completo.

3. Si consideri una funzione booleana $f$ a tre argomenti definita attraverso la seguente tabella di verità:

| $x$ | $y$ | $z$ | $F$ |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |

| $x$ | $y$ | $z$ | $F$ |
|---|---|---|---|
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

Si dimostri dunque che il linguaggio $A \subseteq \{0,1\}^*$ definito come

$$A = \left\{ a_1b_1c_1a_2b_2c_2 \dots a_nb_nc_n \,\middle|\, n \geq 0 \wedge (\forall i \in \{1,\dots,n\}) f(a_i,b_i,c_i) = 1 \right\}$$

è regolare e si determini l'automa minimo che lo accetta.

4. Si mostri che

$$B = \left\{ z \in \{0,1\}^* \,\middle|\, z = 0^{\lfloor \frac{|z|}{2} \rfloor} 1^{\lfloor \frac{|z|}{2} \rfloor} \vee z = 0^{\lfloor \frac{|z|}{4} \rfloor} 1^{\lfloor \frac{|z|}{4} \rfloor} 0^{\lfloor \frac{|z|}{4} \rfloor} 1^{\lfloor \frac{|z|}{4} \rfloor} \right\}$$

non è libero dal contesto. E' decibile?

5. Si studino gli insiemi:

$$\begin{array}{rcl} C & = & \{ x \in \mathbb{N} \mid (\exists k \in \mathbb{N}) \varphi_x(x+k) \downarrow \} \\ D & = & \{ \langle x,y,z \rangle \mid W_x = \{x\} \wedge \emptyset \subset E_y \subset E_z \} \end{array}$$

Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

20 settembre 2006

1. Si enuncino e dimostrino le proprietà di chiusura dei linguaggi liberi dal contesto.
2. Si dimostri l'esistenza di un insieme semplice.
3. Si considerino, al variare di $i \in \mathbb{N}$, i seguenti insiemi:

$$A_i = \left\{ \ 0^n 1^{i \cdot n} \ \middle| \ n \in \mathbb{N} \ \right\}$$

   (a) Si studino, al variare di $i \in \mathbb{N}$, tali insiemi, con particolare attenzione alla proprietà di essere regolare o libero dal contesto.

   (b) Si studi dunque $B = \bigcup_{i \in \mathbb{N}} A_i$.

4. Si dimostri che esiste $n \in \mathbb{N}$ tale che, per ogni $x \in \mathbb{N}$, $\varphi_n(x) = x + n$
5. Si studino gli insiemi:

$$\begin{array}{rcl}
C & = & \left\{ \ x \in \mathbb{N} \ \middle| \ \varphi_x(x+1) \downarrow \wedge \varphi_x(x+2) \downarrow \wedge \cdots \wedge \varphi_x(x+x) \downarrow \ \right\} \\
D & = & \left\{ \ \eta(P) \ \middle| \ \begin{array}{l} \eta \text{ funzione biiettiva ricorsiva dall'insieme dei programmi C a } \mathbb{N}, \\ P \text{ programma C}, \\ [\![P]\!]([v_1, \ldots, v_n]) = [v_{i_1}, \ldots, v_{i_n}], \text{ove } n \geq 0, \\ \text{i vari } v_i \text{ sono } \texttt{int}, \text{ e } v_{i_1} \leq \cdots \leq v_{i_n} \end{array} \right\} \\
E & = & \left\{ \ \langle x, y \rangle \ \middle| \ x \leq y \wedge (\forall z \in \{x, x+1, \cdots, y\})(\varphi_z(z) \uparrow) \ \right\}
\end{array}$$

# Università degli Studi di Udine

Prova Scritta di *Fondamenti dell'Informatica II*

20 Marzo 2007

1. Si enunci il Teorema di Myhill-Nerode e si dimostri una delle tre implicazioni.

2. Si dia la definizione di insieme semplice e si dimostri che un insieme semplice non può essere né ricorsivo né completo.

3. Si dimostri che il seguente insieme $A$ non è libero dal contesto

$$A = \left\{ x + y = z \;\middle|\; \begin{array}{l} x, y, z \text{ sono rappresentazioni binarie senza zeri inutili} \\ \text{di numeri naturali che soddisfano l'equazione} \end{array} \right\}$$

4. Si studino gli insiemi:

$$\begin{array}{rcl} B & = & \{ \langle x, y \rangle \mid W_{x \cdot y} \supseteq \{0, 1\} \} \\ C & = & \{ x \mid 3 \leq |E_x| \leq 5 \} \end{array}$$

5. Si consideri il linguaggio $S \subseteq \{0, 1, \vee, ;\}$ tale che una stringa $x \in L$ se e solo se è una istanza sintatticamente corretta di 3SAT, ove le variabili $X_1, X_2, X_3, \ldots$ sono rappresentate dal numero naturale indicante il pedice, espresso in binario senza zeri inutili $1, 10, 11, \ldots$ e i letterali negati $\neg X_1, \neg X_2, \neg X_3$ da numeri binari come sopra ma preceduti da uno 0: $01, 010, 011, \ldots$, mentre un ";" separa le clausole (non si usa il simbolo $\wedge$). Ad esempio:

$$1 \vee 010 \vee 11; 01 \vee 10 \vee 011; 11 \vee 010 \vee 011$$

   (a) Si dica se $L$ è (o meno) regolare o libero dal contesto

   (b) Sotto l'ipotesi $P \neq NP$ si dica se il sottoinsieme `YES` $\subseteq S$ delle istanze soddisfacibili di 3SAT è regolare o libero dal contesto

# UNIVERSITÀ DEGLI STUDI DI UDINE

Prova Scritta di *Fondamenti dell'Informatica II*

3 Aprile 2007

1. Si dia la definizione di insieme produttivo e si dimostri che tale insieme non può essere né ricorsivo né r.e. Si dimostri inoltre che $\bar{K}$ non è estensionale.

2. Si dimostri che i linguaggi liberi dal contesto sono chiusi per unione e concatenazione ma non per intersezione.

3. Si dimostri che

$$A \;=\; \left\{\, x \in \{0,1\}^* \;\middle|\; (\text{il numero di 0 in } x) \leq 2 \cdot (\text{il numero di 1 in } x) \,\right\}$$

   (a) è libero dal contesto ma (b) non regolare.

4. Sia $\eta$ una funzione calcolabile biiettiva che associa ad ogni programma C un diverso numero naturale (si assuma di saper calcolare anche l'inversa di $\eta$ che dato un numero, restituisce un programma C). Sia $\mathcal{A}$ l'insieme dei caratteri ASCII stampabili e assumiamo che tali programmi accettino in input una stringa (di lunghezza arbitraria) di caratteri di $\mathcal{A}$.

   Sia $\mathcal{P}rim$ l'insieme dei programmi $P$ tali che

$$[\![P]\!](x) = \begin{cases} 1 & \text{Se } x \text{ rappresenta un numero primo espresso in decimale} \\ 0 & \text{Se } x \text{ rappresenta un numero ma non primo espresso in decimale} \\ 2 & \text{Per tutte le altre stringhe} \end{cases}$$

   Si studino i seguenti insiemi:

$$\begin{array}{rcl} B & = & \left\{\, \eta(P) \;\middle|\; P \in \mathcal{P}rim \,\right\} \\ C & = & \left\{\, \eta(P) \;\middle|\; (\exists Q \in \mathcal{P}rim)(\forall x \in \mathcal{A}^*)(|x| < 10^{120} \to [\![P]\!](x) = [\![Q]\!](x)) \,\right\} \end{array}$$

5. Stiamo organizzando un torneo di *calcio a tre* ad $n$ squadre (per semplicità denotate come 1, 2, ..., $n$) in una frazione di Sperdutis, raggiungibile solo da pullman autorizzati. Sappiamo che ci sono delle squadre che si sopportano e che possono essere caricate contemporaneamente sullo stesso pullman (non preoccupatevi della dimensione del pullman, immaginatelo pure enorme). Viceversa, coppie di squadre che non si sopportano non possono essere messe sullo stesso pullman in quanto ne andrebbe di mezzo la riuscita del torneo.

   Sia $f \subseteq \{1, \ldots, n\}^2$ la relazione —simmetrica— di amicizia tra coppie di squadre. Si mostri che, dati $n$, $f$ come sopra e $k \in \mathbb{N}$, stabilire se esiste un modo per posizionare le $n$ squadre in $k$ pullman diversi senza pregiudicare il torneo è un problema NP-completo.

**Università degli Studi di Udine**
Prova Scritta di *Fondamenti dell'Informatica II*
24 Luglio 2007

1. Si enunci e dimostri il primo teorema di ricorsione.
2. Si dimostri che esistono insiemi non r.e. che non sono produttivi.
3. Assumendo Circuit Sat NP completo, di dimostri la NP completezza di 3SAT.
4. Si collochi il linguaggio

$$A = \left\{ \ 0^a 1^b 0^c \,\middle|\, a > 0 \land b > 0 \land c > 0 \land (c = a + b \lor c = a) \ \right\}$$

nella gerarchia di Chomsky. Se non è regolare, lo si dimostri formalmente. Se è CF, si determini inoltre una grammatica in forma normale di Chomsky che lo genera.

5. Si studi l'insieme

$$B = \left\{ \ \langle x, y \rangle \,\middle|\, \exists z\, (\varphi_x(z) \downarrow \land \varphi_y(z) \downarrow \land \varphi_x(z) > \varphi_y(z)) \ \right\}$$

6. Sia $\eta$ una funzione biiettiva dall'insieme dei programmi **C** con due parametri e $\mathbb{N}$. Sia $\mathcal{C}$ l'insieme dei programmi **C** che prendono in input una grammatica CF $G$ sull'alfabeto $\{0, 1\}$, scritta in ASCII seguendo una opportuna notazione (non serve fissarne una) e una stringa $x$ di caratteri in $\{0, 1\}^*$ e restituisce 1 se $x \in L(G)$, 0 se $x \notin L(G)$, -1 se c'è qualche errore sintattico in $G$ o in $x$. Si studi l'insieme:

$$C = \left\{ \ \eta(P) \,\middle|\, P \in \mathcal{C} \ \right\}$$

Università degli Studi di Udine
Prova Scritta di *Fondamenti dell'Informatica II*
5 Settembre 2007

1. Si enunci e dimostri il pumping lemma per i linguaggi liberi dal contesto.

2. Si dimostri che ogni insieme produttivo ha un sottoinsieme ricorsivo ed infinito.

3. Si collochi il linguaggio

$$A = \left\{ 0^n v 0^n \;\middle|\; \begin{array}{l} n \geq 0, v \in \{0,1\}^*, \text{il simbolo più a sinistra di } v \text{ è un 1 e} \\ \text{se } v \text{ è interpretato come numero espresso in binario, è un multiplo di 7} \end{array} \right\}$$

nella gerarchia di Chomsky. Se non è regolare, lo si dimostri formalmente. Se è CF, si definisca una grammatica che lo genera.

4. Si dimostri che esiste $n \in \mathbb{N}$ tale che

$$\varphi_n(n) = 1, \varphi_n(2n) = 2, \varphi_n(3n) = 3, \varphi_n(4n) = 4, \ldots$$

5. Sia $\eta$ una funzione biiettiva ricorsiva dall'insieme dei programmi $\mathbf{C}$ ad un argomento e $\mathbb{N}$. Si studi l'insieme

$$B = \left\{ \eta(P) \;\middle|\; [\![P]\!](\underbrace{1 \cdots 1}_{|P|}) = \sqrt{|P|} \right\}$$

dove $|P|$ denota la dimensione del file sorgente (in bytes).

6. Si studi l'insieme

$$C = \left\{ x \;\middle|\; 1 \leq |E_x| \leq 5 \right\}$$

UNIVERSITÀ DEGLI STUDI DI UDINE
Prova Scritta di *Fondamenti dell'Informatica II*
20 Settembre 2007

1. Si dimostri che i linguaggi liberi dal contesto non sono chiusi per complementazione.

2. Si fornisca con precisione la nozione di completezza per una classe di complessità e si dimostri che il problema SAT sta in NP.

3. Si collochi il linguaggio

$$A = \left\{ \, 0^a 1^b 0^c \;\middle|\; a, b, c \in \mathbb{N} \text{ e } (c = a \text{ oppure } c = 2a) \, \right\}$$

nella gerarchia di Chomsky. Se non è regolare, lo si dimostri formalmente. Se è CF, si definisca inoltre una grammatica che lo genera.

4. Si dimostri che esiste $x \in \mathbb{N}$ tale che $\varphi_x$ calcola la funzione:

$$\begin{array}{rcl} \varphi_x(0) & = & x \\ \varphi_x(1) & = & x - 1 \\ & \vdots & \\ \varphi_x(x) & = & 0 \\ \varphi_x(y) & \uparrow & \text{se } y > x \end{array}$$

5. Sia $\eta$ una funzione biiettiva dall'insieme dei programmi **C** con un parametro e $\mathbb{N}$. Si studi l'insieme:

$$B = \left\{ \, \eta(P) \;\middle|\; (\forall x \in \{0,1\}^*)(|x| < |P| \to [\![P]\!](x) = |x|) \, \right\}$$

ove al solito $|x|$ denota la lunghezza della stringa $x$.

6. Si studi l'insieme

$$C = \left\{ \, \langle x, y \rangle \;\middle|\; (\exists z \in \mathbb{N})(x \leq z \leq y \wedge \varphi_x(z) = 5) \wedge (\exists z \in \mathbb{N})(z > y \wedge \varphi_x(z) \uparrow) \, \right\}$$

UNIVERSITÀ DEGLI STUDI DI UDINE
Prova Scritta di *Fondamenti dell'Informatica II*
27 Marzo 2008

1. Si dia la definizione di insieme produttivo e si dimostri che ogni insieme produttivo ammette un sottoinsieme *ricorsivo* e infinito.

2. Si fornisca con precisione la nozione di completezza per una classe di complessità e si dimostri che il problema SAT sta in NP.

3. Si denotino con $p_1 = 2, p_2 = 3, p_3 = 5, \ldots$ i numeri primi.

   (a) Si mostri che, fissato $i \in \mathbb{N}$, il linguaggio
   $$A_i = \left\{ \; 0^{p_j} 1 0^{p_k} v \;\middle|\; v \in \{0,1\}^*, p_j < p_k, j \leq i, k \leq i \; \right\}$$
   è regolare.

   (b) Si fornisca il DFA minimo che accetta $A_2$

   (c) Si dimostri che $B = \bigcup_{i \in \mathbb{N}} A_i$ non è libero dal contesto.

4. Si collochi l'insieme $C$ nella gerarchia della computabilità:
$$C = \left\{ \; x \;\middle|\; \varphi_{\lfloor \frac{x}{3} \rfloor}(3x^3) = x + 3 \; \right\}$$

5. Sia $\eta$ una funzione biiettiva dall'insieme dei programmi **while** con un parametro e $\mathbb{N}$. Sia $\mathcal{L}$ l'insieme delle liste con elementi in $\{A, T, C, G\}$ (ad esempio $(A.(T.(C.\mathtt{nil})))$ che si può anche denotare come $(A, T, C)$). Si studi l'insieme:
$$D = \left\{ \; \eta(P) \;\middle|\; (\forall x \in \mathcal{L})(|x| < 1024 \to [\![P]\!](x) = \#(A, x) \; \right\}$$
ove $|x|$ denota la lunghezza della lista $x$ e $\#(A, x)$ il numero di occorrenze del simbolo $A$ nella lista $x$.

6. Si studi l'insieme
$$E = \left\{ \; \eta(P) \;\middle|\; (\forall x \in \mathcal{L})([\![P]\!](x) = \#(A, x)) \; \right\}$$

Università degli Studi di Udine
Prova Scritta di *Fondamenti dell'Informatica II*
15 Aprile 2008

1. Si enunci e dimostri il primo teorema di ricorsione.
2. Si dica se valgono o meno (e perché) le riduzioni $K \preceq \bar{K}$ o $\bar{K} \preceq K$.
3. Si considerino, al variare di $i \in \mathbb{N}$ gli insiemi:
$$L_i = \left\{ \, 0^x 1^y \;\middle|\; x \leq i, y \leq x^2 \, \right\}$$
   (a) Si mostri che, fissato $i \in \mathbb{N}$, $L_i$ è regolare.
   (b) Si dica (motivando la risposta) se $L = \bigcup_{i>0} L_i$ è o meno libero dal contesto
   (c) Si dica (motivando la risposta) se $L$ è decidibile.
4. Si collochino i seguenti insiemi nella gerarchia della ricorsione:
$$\begin{array}{rcl} A & = & \left\{ \, x \;\middle|\; \varphi_x(0) = 0 \wedge \varphi_x(10) = 1 \wedge \varphi_x(100) = 2 \wedge \cdots \wedge \varphi_x(10^x) = x \, \right\} \\ B & = & \left\{ \, \langle x, y \rangle \;\middle|\; \varphi_x(y) \downarrow \wedge \varphi_y(x) \downarrow \wedge (\forall z \in \mathbb{N})((\varphi_x(z) \downarrow \wedge \varphi_y(z) \downarrow) \rightarrow \varphi_x(z) = 1 + \varphi_y(z)) \, \right\} \\ C & = & \left\{ \, x \;\middle|\; W_x = \{x!\} \, \right\} \end{array}$$
5. Sia 4SAT il problema di stabilire se una formula logica congiunzione di clausole di esattamente 4-letterali sia o meno soddisfacibile. Si dimostri la sua NP-completezza, senza usare l'informazione che 3SAT è NP-completo.

UNIVERSITÀ DEGLI STUDI DI UDINE
Prova Scritta di *Fondamenti dell'Informatica II*
27 Giugno 2008

1. Si enuncino e dimostrino le proprietà di chiusura dei linguaggi liberi dal contesto.
2. Si dimostri che un insieme semplice non è né ricorsivo né completo.
3. Si consideri il linguaggio
$$A = \left\{ \; 10^i 1 \;\middle|\; i \text{ è pari e non è un multiplo di 3 } \right\}$$
   (a) Si mostri che è regolare
   (b) Si determini il DFA minimo che lo accetta
4. Si mostri che il seguente linguaggio non è libero dal contesto
$$A = \left\{ \; 0^m 1^{m^m} \;\middle|\; m > 0 \; \right\}$$
5. Sia $n$ l'indice di una MdT *totale*. Si studi l'insieme:
$$C = \left\{ \; x \;\middle|\; \varphi_x(x) = \varphi_n(x) \; \right\}$$
   E se rimuovessimo l'ipotesi di totalità?
6. Si studi l'insieme:
$$D = \left\{ \; \langle x, y \rangle \;\middle|\; \{0,1\} \subseteq W_x \subset W_y \subseteq \{0,1,2,3\} \; \right\}$$
7. Si consideri il seguente problema. *Sono dati $n$ oggetti di peso $p_1, p_2, p_3, \ldots, p_n$ e $k$ secchi che possono portare (ciascuno) al più peso $p$. Esiste un modo per sistemare tutti gli oggetti nei secchi in modo tale da rispettare i vincoli di peso per ogni secchio?*

   Si dimostri che il problema sta in NP. Si discuta (informalmente o se avete tempo formalmente) sulla sua eventuale completezza.

**Università degli Studi di Udine**
Prova Scritta di *Fondamenti dell'Informatica II*
16 Luglio 2008

1. Si dimostri che l'insieme dei linguaggi accettati da NFA coincide con quello dei linguaggi accettati da DFA.

2. Si dia la definizione di insieme produttivo e si dimostri che ogni insieme produttivo ammette un sottoinsieme r.e. infinito.

3. Sapendo che SAT è NP completo si dimostri con precisione che 3SAT è NP completo.

4. Si studi il linguaggio
$$A = \left\{ a^n b^{n+2} a^m c^{n+3} \;\middle|\; m, n \geq 0 \right\}$$
In particolare si dica (dimostrandolo) se è o meno regolare, libero dal contesto, decidibile.

5. Si mostri che esiste $x \in \mathbb{N}$ tale che $W_x = \{0, 2, 4, \ldots, 2x\}$.

6. Sia $\eta$ una funzione biiettiva dall'insieme dei programmi **C** con un parametro e $\mathbb{N}$. Sia $\mathcal{L}$ l'insieme delle liste di numeri `int`. Si studino gli insiemi:
$$\begin{array}{rcl} B & = & \left\{ \eta(P) \;\middle|\; (\forall x \in \mathcal{L})(|x| \leq 32 \rightarrow \llbracket P \rrbracket(x) = \min(x) \right\} \\ C & = & \left\{ \eta(P) \;\middle|\; (\forall x \in \mathcal{L})(|x| > 32 \rightarrow \llbracket P \rrbracket(x) = \min(x) \right\} \end{array}$$
ove $|x|$ denota la lunghezza della lista $x$ e $\min(x)$ il minimo intero tra gli elementi della lista $x$.

# Università degli Studi di Udine
## Prova Scritta di *Fondamenti dell'Informatica II*
## 25 Settembre 2008

1. Si dimostri l'esistenza e l'unicità del DFA minimo per un linguaggio regolare.
2. Si dimostri che esistono insiemi non r.e. che non sono produttivi.
3. Si dimostri che il seguente insieme non è libero dal contesto:

$$A = \left\{ 0^x 1^{x^2} 0^y \,\middle|\, \{x, y\} \subseteq \mathbb{N} \right\}$$

   Si dica inoltre se è ricorsivo e, nel caso, in quale classe di complessità temporale vada collocato.

4. Si dimostri che esiste $x \in \mathbb{N}$ tale che $W_x = \{x, 2x, 3x, 4x, \ldots\}$
5. Si studino (si dica se sono estensionali, r.e., ricorsivi, completi, produttivi, semplici etc.) gli insiemi:

$$\begin{array}{rcl} B & = & \{ x \mid (\forall i \in \{0, \ldots, x\})(\varphi_x(i) = i) \} \\ C & = & \{ \langle x, y \rangle \mid W_x \subseteq E_y \} \end{array}$$

   Nel caso non siano r.e., si studino anche i loro complementari.

# UNIVERSITÀ DEGLI STUDI DI UDINE

Prova Scritta di *Fondamenti dell'Informatica II*

10 Febbraio 2009

1. Si enunci e dimostri il pumping lemma per i linguaggi liberi dal contesto.

2. Si definisca la nozione di insieme semplice e si dimostri che un insieme semplice non è né ricorsivo né creativo.

3. Si studi il linguaggio

$$A = \left\{ 1^m 0^n 1^p \;\middle|\; m, n, p \geq 0,\, p = m + n \text{ oppure } p = m - n \right\}$$

   Qualora sia regolare si determini il DFA minimo che lo accetta. Qualora non sia regolare, lo si dimostri. Similmente, qualora sia libero dal contesto si scriva una grammatica che lo genera. Qualora non lo sia, lo si dimostri.

4. Si mostri che esiste $\nu \in \mathbb{N}$ tale che

$$W_\nu = E_\nu = \{0, 1, 2, 3, \ldots, \nu\}$$

5. Si studino i seguenti insiemi ed i loro complementari (in $C$ se entrambe sono indefinite, vanno considerate uguali):

$$\begin{array}{rcl} B & = & \left\{ x \;\middle|\; \varphi_x(x)\downarrow \wedge \varphi_x(2x)\downarrow \wedge \varphi_x(x) = \varphi_x(2x) \right\} \\ C & = & \left\{ x \;\middle|\; \varphi_x(x) = \varphi_x(2x) \right\} \\ D & = & \left\{ \langle x, y\rangle \;\middle|\; (\forall z \leq x + y)(\varphi_z(x)\downarrow \wedge \varphi_z(y)\downarrow) \right\} \end{array}$$

6. Si consideri il seguente problema. *Sono dati $n$ oggetti di costo $p_1, p_2, p_3, \ldots, p_n$ e $k$ persone che possiedono euro $a_1, \ldots, a_k$ rispettivamente. Esiste un modo per vendere tutti gli $n$ oggetti alle varie persone in modo tale che nessuno di essi si debba indebitare?*

   Si dimostri che il problema sta in NP. Si discuta (almeno informalmente) sulla sua eventuale completezza.