Informatica Teorica - Compito A

7 luglio 2003

Cognome:...
Nome: ....
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Vertex Cover (VC): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$ esiste un vertex cover $V' \subseteq V$, $|V'| \leq k$, tale che, per ogni arco $e = (v_1, v_2) \in E$, almeno uno tra v_1 e v_2 sia contenuto in V'.
   Cupola Mafiosa (CM): Un boss mafioso ha alle sue dipendenze alcuni capi cosca. Ogni cosca contiene un certo numero di picciotti. Un picciotto può appartenere a più cosche. Può il boss mandare un ordine a tutti i piccotti coinvolgendo al più k capi cosca?

   Premesso che il problema VC è noto essere NP-completo, mostrare che anche CM è NP-completo.

2. Scrivere un programma LISP che, ricevuti in ingresso una lista di liste, restituisca la lista contenente l'intersezione delle liste.
   Esempio: se `L=((A B C) (B C) (B C D))` il programma deve restituire `L'=(B C)`.

3. Dimostrare che le funzioni Turing calcolabili sono calcolabili mediante RAM e illustrare il costo della simulazione sia nel modello a costi logaritmici sia nel modello a costi unitari.

4. Dimostrare che esiste un programma di codice i tale che:

$$f_i(x) = i^2$$

Informatica Teorica - Compito B

7 luglio 2003

Cognome:...
Nome: ....
Matricola:..

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Cupola Mafiosa (CM): Un boss mafioso ha alle sue dipendenze alcuni capi cosca. Ogni cosca contiene un certo numero di picciotti. Un picciotto può appartenere a più cosche. Può il boss mandare un ordine a tutti i piccotti coinvolgendo al più k capi cosca?
   Hitting Set (HS): Dati un insieme S, m sottoinsiemi $C_1, C_2, \ldots, C_m$ di 2^S e una costante k, decidere se esiste un sottoinsieme $S' \subseteq S$ tale che $|S'| \leq k$ e S' contiene almeno un elemento di ciascun $C_i, i = 1, 2, \ldots, m$.

   Premesso che il problema CM è noto essere NP-completo, mostrare che anche HS è NP-completo.

2. Scrivere un programma LISP che, ricevuti in ingresso una lista di liste, restituisca la lista contenente l'unione delle liste.
   Esempio: se `L=((A B C) (B C) (B C D))` il programma deve restituire `L'=(A B C D)`.

3. Dimostrare che le funzioni calcolabili mediante RAM sono Turing calcolabili e illustrare il costo della simulazione nel modello a costi logaritmici.

4. Dimostrare che esiste un programma di codice i tale che:

$$f_i(x) = i \cdot x$$

Informatica Teorica - Compito B

15 luglio 2003

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Bounded Degree Spanning Tree (BDST): Dato un grafo $G = (V, E)$ e una costante $k > 0$ esiste un albero ricoprente in cui il grado di ogni vertice é inferiore o uguale a k?
   Cammino Hamiltoniano (HP): Dato un grafo $G = (V, E)$ esiste un cammino che tocca tutti i vertici una ed una sola volta?

   Mostrare che $HP \leq BDST$.

2. Si consideri il seguente problema.

   Hitting Set (HS): Dati un insieme S, m sottoinsiemi $C_1, C_2, \ldots, C_m$ di 2^S e una costante k, decidere se esiste un sottoinsieme $S' \subseteq S$ tale che $|S'| \leq k$ e S' contiene almeno un elemento di ciascun $C_i, i = 1, 2, \ldots, m$.

   Scrivere un programma LISP che, ricevute in ingresso una lista L di sottoinsiemi $C_1, C_2, \ldots, C_m$ (rappresentati come liste) e una lista che rappresenta S', verifichi se S' è un hitting set di L. Per esempio, se $L =$`((A B) (B C D) (E F))` e $S' =$`(B E)`, allora S' è un hitting set di L.

3. Enunciare e dimostrare le principali proprietá degli insiemi ricorsivamente enumerabili.

4. Illustrare la simulazione della Macchina di Turing Multinastro mediante la Macchina di Turing a un nastro.

Informatica Teorica - Compito A

15 luglio 2003

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Set Packing (SP): Data una collezione C di insiemi finiti e una costante $k \leq |C|$, C contiene almeno k insiemi disgiunti?
   Independent Set (IS): Dato un grafo $G = (V, E)$ e una costante $k \geq 0$ esiste $V' \subseteq V$, con $|V'| = k$, e tale che per ogni coppia $u, v \in V'$ $(u, v) \notin E$?

   Mostrare, a scelta, che $IS \leq SP$ o che $SP \leq IS$.

2. Si consideri il seguente problema.

   Set Cover (SC): Dati un insieme S, una collezione C di sottoinsiemi di S $C = \{c_1, c_2, \ldots, c_m\}$ e una costante k, decidere se esiste un sottoinsieme $C' \subseteq C$ tale che $|C'| \leq k$ e C' copre S, ovvero $S = \cup_{c_i \in C'}$.

   Scrivere un programma LISP che, ricevute in ingresso una lista L rappresentante l'insieme S e una lista C' di sottoinsiemi $c_1, c_2, \ldots, c_m$ (rappresentati come liste) verifichi se C' è un set cover di L. Per esempio, se $L =$`(A B C D E F)` e $C' =$`((A B C)(B E F)(D E))`, allora C' è un set cover di L.

3. Dimostrare che i problemi $\overline{K} = \{ \ x \mid \varphi_x(x) \text{ non è definita}\}$ e $T = \{ \ x \mid \varphi_x \ \text{ una funzione totale } \}$ non sono ricorsivamente enumerabili.

4. Illustrare la simulazione della Macchina di Turing Non Deterministica mediante la Macchina di Turing Deterministica.

Informatica Teorica - Compito A

9 settembre 2003

Cognome:..
Nome:..
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Isomorfismo di Sottografo (IS): Dati due grafi $G = (V_1, E_1)$ e $H = (V_2, E_2)$, G contiene un sottografo isomorfo a H? Ovvero, esistono due insiemi $V \subseteq V_1$ e $E \subseteq E_1$ tali che $|V| = |V_2|$, $|E| = |E_2|$ e una funzione biunivoca $f : V_2 \rightarrow V$ tale che, se l'arco $\{u, v\} \in E_2$ allora l'arco $\{f(u), f(v)\} \in E$?

   Cricca (C): Dato un grafo G=(V,E) e una costante k, G contiene una cricca di dimensione k? Ovvero, esiste un insieme $V' \subseteq V$, $|V'| \geq k$, tale che per ogni $u, v \in V'$ l'arco $\{u, v\} \in E$?

   Mostrare che $C \leq IS$.

2. Scrivere un programma LISP che, ricevute in ingresso una lista di liste L, rappresentante una matrice quadrata di dimensione $n \times n$, restituisca una lista contenente gli elementi sulla diagonale *principale*. Per esempio, se $L =$ `((A B B) (B C D) (E F F))`, il programma deve restituire la lista `(A C F)`.

3. Dimostrare che gli insiemi r.e. sono chiusi rispetto all'unione ma non rispetto alla complementazione.

4. Come si dimostra che ogni linguaggio di tipo 0 può essere accettato da una macchina di Turing?

Informatica Teorica - Compito B

9 settembre 2003

Cognome: ....
Nome: ....
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Isomorfismo di Sottografo (IS): Dati due grafi $G = (V_1, E_1)$ e $H = (V_2, E_2)$, G contiene un sottografo isomorfo a H? Ovvero, esistono due insiemi $V \subseteq V_1$ e $E \subseteq E_1$ tali che $|V| = |V_2|$, $|E| = |E_2|$ e una funzione biunivoca $f : V_2 \to V$ tale che, se l'arco $\{u, v\} \in E_2$ allora l'arco $\{f(u), f(v)\} \in E$?

   Circuito Hamiltoniano (CH): Dato un grafo G=(V,E), G contiene un circuito Hamiltoniano? Ovvero, esiste un cammino che tocca tutti i vertici una ed una sola volta?

   Mostrare che $CH \leq IS$.

2. Scrivere un programma LISP che, ricevute in ingresso una lista di liste L, rappresentante una matrice quadrata di dimensione $n \times n$, restituisca una lista contenente gli elementi sulla diagonale *secondaria*. Per esempio, se $L =$`((A B B) (B C D) (E F F))`, il programma deve restituire la lista `(B C E)`.

3. Definire una funzione totale che enumera l'insieme $K = \{\ x \mid \varphi_x(x) \text{ è definita}\}$.

4. Come si dimostra che ogni linguaggio accettato da una macchina di Turing è di tipo 0?

Informatica Teorica - Compito A

12 gennaio 2004

Cognome:...
Nome: ....
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Traveling Salesman Problem (TSP): Dato un intero $k \geq 0$ e un grafo completo $G = (V, E)$, dove ad ogni arco e_i è associato un peso intero w_i, decidere se esiste un ciclo Hamiltoniano in cui la somma dei pesi sugli archi sia inferiore a k.
   Quota-Traveling Salesman Problem (Q-TSP): Dato un intero $k \geq 0$, un intero $Q \geq 0$ e un grafo completo $G = (V, E)$, dove ad ogni arco e_i è associato un peso intero w_i e a ogni nodo è associato un peso intero q_j, decidere se esiste un ciclo in cui la somma dei pesi sugli archi sia inferiore a k e la somma dei pesi sui nodi sia maggiore di Q.

   Premesso che il problema TSP è noto essere NP-completo, mostrare che anche il problema Q-TSP è NP-completo.

2. Scrivere un programma LISP che, ricevute in ingresso due liste rappresentanti insiemi, restituisca la lista rappresentante l'insieme ottenuto come prodotto cartesiano dei due insiemi.
   Esempio: se `L1=(A B C)` e `L2=(1 2)` il programma deve restituire
   `L'=((A 1)(A 2)(B 1)(B 2)(C 1)(C 2))` (non necessariamente in quest'ordine).

3. Dimostrare che la classe degli insiemi semidecidibili contiene strettamente la classe degli insiemi decidibili.

4. Definire la classe dei problemi di decisione NP-completi e spiegare, anche con un esempio, come si dimostra che un problema è NP-completo.

Informatica Teorica - Compito B

12 gennaio 2004

Cognome:. . . . .
Nome: . . . .
Matricola: . . . .

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Si considerino i seguenti problemi:

   Cammino Hamiltoniano (HP): Dato un grafo G=(V,E), G contiene un cammino Hamiltoniano? Ovvero, esiste un cammino che tocca tutti i vertici una ed una sola volta?
   K-Path (KP): Dato un intero $K \geq 0$ e un grafo completo $G = (V, E)$, dove ad ogni arco e_i è associato un peso intero w_i, decidere se esiste un cammino in cui la somma dei pesi sugli archi sia superiore a K.

   Premesso che il problema HP è noto essere NP-completo, mostrare che anche il problema KP è NP-completo.

2. Scrivere un programma LISP che, ricevute in ingresso due liste rappresentanti insiemi, restituisca la lista rappresentante l'insieme ottenuto come differenza simmetrica dei due insiemi.
   Esempio: se `L1=(A B C D)` e `L2=(E D C F)` il programma deve restituire `L'=(A B E F)`. (Ricordiamo che $\Delta(A, B) = A \bigcup B - A \bigcap B$).

3. Dimostrare che la classe degli insiemi ricorsivi è strettamente inclusa nella classe degli insiemi ricorsivamente enumerabili.

4. Definire le classe P, NP, PSPACE ed EXPTIME , mostrare quali relazioni di inclusione tra di esse sono note e spiegarne i motivi.

Informatica Teorica - Compito A

25 marzo 2004

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Illustrare e discutere le relazioni esistenti tra la classi di complessità (spaziali e temporali) deterministiche e nondeterministiche.

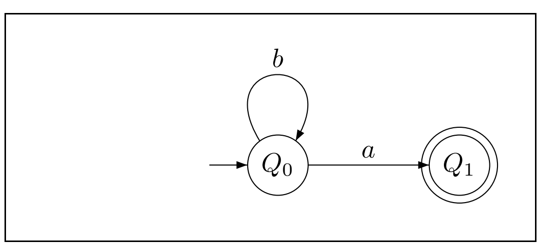
2. Come si può dimostrare che la ricursione primitiva e la composizione non consentono di definire tutte le funzioni calcolabili secondo Turing?

3. Si considerino i seguenti problemi:
   Graph K-Colorability (GKC): Dato un intero $k \geq 0$ e un grafo $G = (V, E)$, decidere se è possibile colorare i vertici con k colori. Ovvero, esiste una funzione $f : V \rightarrow \{1, 2 \dots k\}$ tale che, per ogni arco $(u, v) \in E$, si ha $f(u) \neq f(v)$?
   Graph K-Colorability con 2 Vertici Fissati(GKC2VF): Dato un intero $k \geq 0$, un grafo $G = (V, E)$ e 2 vertici $v_1, v_2 \in V$ non adiacenti, decidere se è possibile colorare i vertici con k colori con il vincolo che v_1 e v_2 abbiano lo stesso colore. Ovvero, esiste una funzione $f : V \rightarrow \{1, 2 \dots k\}$ tale che $f(v_1) = f(v_2)$ e, per ogni arco $(u, v) \in E$, si ha $f(u) \neq f(v)$?

   Premesso che il problema GKC è noto essere NP-completo, mostrare che anche il problema GKC2VF è NP-completo.

4. Si consideri il linguaggio L accettato da un automa a stati finiti A. Sia $\delta : Q \times \Sigma \rightarrow Q$ la funzione di transizione dell'automa, implementata in in LISP dal comando `DELTA` (già esistente); per esempio, nell'automa in figura, $\delta(q_0, b) = q_0$ e $\delta(q_0, a) = q_1$. Inoltre la funzione LISP `FINALE` prende in ingresso uno stato e restituisce `T` se lo stato è finale. Considerando l'automa in figura, `(DELTA 'Q0 'b)` restituisce `Q0`, `(FINALE 'Q1)` restituisce `T`.

   Scrivere un programma LISP `ASF` che, ricevuto in ingresso lo stato iniziale di un automa e una lista contente gli elementi della stringa da riconoscere, restituisca `T` se la stringa appartiene al linguaggio L. Per esempio, per l'automa rappresentato in figura, `(ASF 'Q0 '(b b b a))` deve restituire `T`.



Informatica Teorica - Compito B

25 marzo 2004

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Se un problema è risolubile da una MT a più nastri in tempo $t(n)$ e spazio $s(n)$, cosa possiamo dire per spazio e tempo utilizzati da una MT ad un nastro per risolvere lo stesso problema?

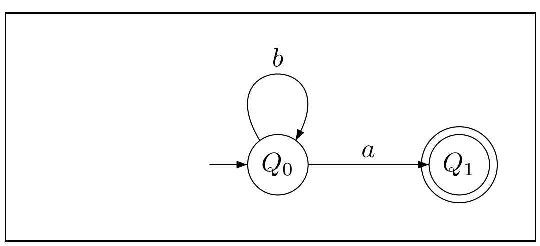
2. Come si può dimostrare che avendo a disposizione istruzioni di tipo condizionale (if-then-else e simili) e ricursione si possono calcolare tutte le funzioni calcolabili secondo Turing?

3. Si considerino i seguenti problemi:
   Hitting Set (HS): Dato un intero $k \geq 0$, un insieme S e una collezione C di sottoinsiemi di S, esiste un Hitting Set di cardinalità k? Ovvero, esiste un insieme $S' \subset S$, $|S'| \leq k$, tale che per ogni $c_i \in C$ si ha che $S' \cap c_i \neq \emptyset$?
   Hitting Set con Elemento Fissato (HSEF): Dato un intero $k \geq 0$, un insieme S, una collezione C di sottoinsiemi di S e un elemento $e \in S$, esiste un Hitting Set di cardinalità k che contiene e? Ovvero, esiste un insieme $S' \subset S$, $|S'| \leq k$, $e \in S'$, tale che per ogni $c_i \in C$ si ha che $S' \cap c_i \neq \emptyset$?

   Premesso che il problema HS è noto essere NP-completo, mostrare che anche il problema HSEF è NP-completo.

4. Si consideri il linguaggio L accettato da un automa a stati finiti A. Sia $\delta : Q \times \Sigma \to Q$ la funzione di transizione dell'automa, implementata in in LISP dal comando `DELTA` (già esistente); per esempio, nell'automa in figura, $\delta(q_0, b) = q_0$ e $\delta(q_0, a) = q_1$. Inoltre la funzione LISP `FINALE` prende in ingresso uno stato e restituisce `T` se lo stato è finale. Considerando l'automa in figura, `(DELTA 'Q0 'b)` restituisce `Q0`, `(FINALE 'Q1)` restituisce `T`.

   Scrivere un programma LISP `ASF` che, ricevuto in ingresso lo stato iniziale di un automa e una lista contente gli elementi della stringa da riconoscere, restituisca `T` se la stringa appartiene al linguaggio $\bar{L}$, **complemento** di L. Per esempio, per l'automa rappresentato in figura, `(ASF 'Q0 '(b b b))` deve restituire `T`.



Informatica Teorica - Compito A

16 aprile 2004

Cognome:. . . . .
Nome:. . . . .
Matricola: . . . .

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Dimostrare che se un insieme S è semidecidibile esso è anche ricorsivamente enumerabile.

2. Dimostrare che i linguaggi accettati da Macchine di Turing deterministiche sono generabili con grammatiche di tipo 0.

3. Si considerino i seguenti problemi:
   Graph K-Colorability (GKC): Dato un intero $k \geq 0$ e un grafo $G = (V, E)$, decidere se è possibile colorare i vertici con k colori. Ovvero, esiste una funzione $f : V \to \{1, 2 \ldots k\}$ tale che, per ogni arco $(u, v) \in E$, si ha $f(u) \neq f(v)$?
   Graph K-Colorability con 2 Vertici Fissati(GKC2VF): Dato un intero $k \geq 0$, un grafo $G = (V, E)$ e 2 vertici $v_1, v_2 \in V$ non adiacenti, decidere se è possibile colorare i vertici con k colori con il vincolo che v_1 e v_2 abbiano lo stesso colore. Ovvero, esiste una funzione $f : V \to \{1, 2 \ldots k\}$ tale che $f(v_1) = f(v_2)$ e, per ogni arco $(u, v) \in E$, si ha $f(u) \neq f(v)$?

   Mostrare che $GKC2VF \leq GKC$, ovvero trasformare una istanza del problema $GKC2VF$ $I_{GKC2VF} = (G, k, v_1, v_2)$ in una istanza di GKC $I_{GKC} = (G', k')$ e mostrare che, se esiste la soluzione S_{GKC}, corrispondente a I_{GKC}, ciò implichi l'esistenza della soluzione S_{GKC2VF}.

4. Scrivere un programma LISP `GKC` che, ricevute in ingresso la descrizione di un grafo (rappresentato come lista di archi) e una colorazione di vertici (rappresentata come lista di coppie vertice colore), restituisca T se la colorazione è una colorazione valida, ovvero se non esistono due vertici adiacenti dello stesso colore, e NIL altrimenti.

   Per esempio, se l'input è costituito dalla lista di archi `L1=((A B) (B C) (C D) (D A))` e dalla lista di colori `L2=((A 1) (B 2) (C 1) (D 2))` il programma dovrebbe restituire T.

Informatica Teorica - Compito B

16 aprile 2004

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Dimostrare che gli insiemi $T = \{x \mid \varphi_x(x) \text{ è totale}\}$ e $\overline{K} = \{x \mid \varphi_x(x) \text{ non è definita}\}$ non sono ricorsivamente enumerabili.

2. Dimostrare che i linguaggi di tipo 0 sono accettati da Macchine di Turing deterministiche. Dimostrare che i linguaggi di tipo 1 sono riconosciuti da Macchine di Turing deterministiche.

3. Si considerino i seguenti problemi:
   Hitting Set (HS): Dato un intero $k \geq 0$, un insieme S e una collezione C di sottoinsiemi di S, esiste un Hitting Set di cardinalità k? Ovvero, esiste un insieme $S' \subset S$, $|S'| \leq k$, tale che per ogni $c_i \in C$ si ha che $S' \cap c_i \neq \emptyset$?
   Hitting Set con Elemento Fissato (HSEF): Dato un intero $k \geq 0$, un insieme S, una collezione C di sottoinsiemi di S e un elemento $e \in S$, esiste un Hitting Set di cardinalità k che contiene e? Ovvero, esiste un insieme $S' \subset S$, $|S'| \leq k$, $e \in S'$, tale che per ogni $c_i \in C$ si ha che $S' \cap c_i \neq \emptyset$?

   Mostrare che $HSEF \leq HS$, ovvero trasformare una istanza del problema $HSEF$ $I_{HSEF} = (S, C, e, k)$ in una istanza di HS $I_{HS} = (S', C', k')$ e mostrare che, se esiste la soluzione S_{HS}, corrispondente a I_{HS}, ciò implichi l'esistenza della soluzione S_{HSEF}.

4. Un grafo *orientato* può essere rappresentato come una lista di archi orientati (a loro volta rappresentati come coppie *ordinate* di nodi).

   Scrivere un programma Lisp che, ricevuti in ingresso un grafo e una lista di nodi, verifichi se tale lista è un cammino. Per esempio, dati `G=((A B)(A C)(B C))`, `L1=(A B C)` e `L2=(B C A)` si dovrebbe avere `(CAMMINO G L1)`$=T$ e `(CAMMINO G L2)`$=NIL$.

Informatica Teorica - Compito A

22 settembre 2004

Cognome:...
Nome:..
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Dimostrare che la classe degli insiemi semidecidibili contiene strettamente la classe degli insiemi decidibili.

2. Come si dimostra che ogni linguaggio di tipo 0 può essere accettato da una macchina di Turing?

3. Si considerino i seguenti problemi:
   Feedback Vertex Set (FVS): dato un grafo diretto $G = (V, E)$ e un intero $k \geq 0$, decidere se esiste un feedback vertex set di cardinalità k. Ovvero, esiste un insieme $V' \subset V$, $|V'| = k$, tale che V' contiene almeno un nodo per ogni ciclo diretto del grafo?
   Feedback Vertex Set con Vertice Fissato (FVSVF): dato un grafo diretto $G = (V, E)$, un intero $k \geq 0$ e un vertice $v \in V$, decidere se esiste un feedback vertex set di cardinalità k contenente v. Ovvero, esiste un insieme $V' \subset V$, $|V'| = k$, $v \in V'$, tale che V' contiene almeno un nodo per ogni ciclo diretto del grafo?

   Premesso che il problema FVSVF è noto essere NP-completo, mostrare che anche il problema FVS è NP-completo.

4. Scrivere un programma LISP `FVS` che, ricevuti in ingresso la lista `L` di tutti i cicli di un grafo diretto e una lista V' contenente un insieme di vertici, restituisca `T` se i vertici di `V'` costituiscono un feedback vertex set. La lista dei cicli è costituita da una lista per ogni ciclo del grafo in cui il primo elemento è il ciclo stesso e gli altri elementi sono i nodi del ciclo; per esempio, se il grafo ha quattro nodi `(A B C D)` e cinque archi diretti `((A B)(B C)(C A)(C D)(D C))`, allora il grafo ha tre cicli rappresentati come `((1 A B C D C)(2 A B C)(3 C D))`. Con questa lista di cicli `L` e `V'=(C)` in input, il programma deve restituire `T`.

Informatica Teorica - Compito B

22 settembre 2004

Cognome:...
Nome:...
Matricola:...

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Dimostrare che la classe degli insiemi ricorsivi è strettamente inclusa nella classe degli insiemi ricorsivamente enumerabili.

2. Come si dimostra che ogni linguaggio accettato da una macchina di Turing è di tipo 0?

3. Si considerino i seguenti problemi:
   Dominating Set (DS): Dato un intero $k \geq 0$ e un grafo $G = (V, E)$, decidere se esiste un dominating set di cardinalità k. Ovvero, esiste un sottoinsieme $V' \subset V$, $|V'| = k$, tale che per ogni $u \in V - V'$ esiste almeno un $v \in V'$ tale che $(u, v) \in E$?
   Dominating Set con Vertice Fissato (DSVF): Dato un intero $k \geq 0$, un grafo $G = (V, E)$ e un vertice $w \in V$, decidere se esiste un dominating set di cardinalità k che includa il vertice w. Ovvero, esiste un sottoinsieme $V' \subset V$,$|V'| = k$, tale che $w \in V'$ e per ogni $u \in V - V'$ esiste almeno un $v \in V'$ tale che $(u, v) \in E$?

   Premesso che il problema DSVF è noto essere NP-completo, mostrare che anche il problema DS è NP-completo.

4. Scrivere un programma LISP `DS` che, ricevuti in ingresso un grafo (rappresentato mediante lista di adiacenze `L`) e una lista V' contenente un insieme di vertici, restituisca `T` se i vertici di `V'` costituiscono un dominating set. La lista di adiacenze è costituita da una lista per ogni nodo del grafo in cui il primo elemento è il nodo stesso e gli altri sono i nodi ad esso adiacenti; per esempio, se il grafo ha cinque nodi `(A B C D E)` e cinque archi `((A B)(A C)(B C)(C D)(D E))` la lista di adiacenze è la seguente: `L=((A B C) (B A C)(C A B D) (D C E)(E D))`. Con questa lista di adiacenze `L` e `V'=(C E)` in input, il programma deve restituire `T`.

Modelli e Complessità di Calcolo - Compito A

5 aprile 2005

Cognome:..
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire la Macchina di Turing non deterministica (MTND). Definire il concetto di accettazione e rifiuto di stringhe da parte di una MTND. Quanto tempo richiede una MTND con un unico nasto per riconoscere il linguaggio $\{ww^R | w \in \{a, b\}^+\}$?

2. Definire i linguaggi ricorsivi e ricorsivamente enumerabili. Dimostrare che il linguaggio $T = \{x \mid \varphi_x$ è una funzione ricorsiva totale $\}$ non è ricorsivamente enumerabile. Definire la Gerarchia Aritmetica. A che classe della Gerarchia Aritmetica appartiene T?

3. Si considerino i seguenti problemi:
   Hitting Set (HS): Dati un insieme S, m sottoinsiemi $C_1, C_2, \ldots, C_m$ di S e una costante k, decidere se esiste un sottoinsieme $S' \subseteq S$ tale che $|S'| \leq k$ e S' contiene almeno un elemento di ciascun $C_i, i = 1, 2, \ldots, m$.
   Accesso ai Docenti (AD): le pagine web dei docenti della Facoltà di Ingegneria sono raggiungibili sia dalle pagine dei rispettivi dipartimenti che dalle pagine dei corsi di Laurea (nell'ambito dei quali i docenti svolgono la loro attività didattica). Dato un intero $k \geq 1$, il problema consiste nel decidere se è possibile creare una pagina web con al più k link (a Dipartimenti o Corsi di Laurea) che consenta di raggiungere tutti i docenti.

   Sapendo che il problema HS è NP-completo, mostrare che anche il problema AD è NP-completo.

4. Relativamente al problema AD definito sopra, scrivere un programma LISP che, presa in ingresso una lista di docenti D e una lista L, contenente liste di nomi di docenti, verifichi se, attraverso le liste in L è possibile accedere a tutti i docenti nella lista D. Per esempio, se $D = \{Aiello\ Ausiello\ Baldoni\ Cadoli\ Lenzerini\}$ e $L = \{\{Aiello\ Baldoni\}\ \{Ausiello\ Baldoni\ Cadoli\}\ \{Ausiello\ Lenzerini\}\}$ il programma deve restituire `T`; se $L = \{\{Aiello\ Cadoli\}\ \{Ausiello\ Cadoli\}\ \{Ausiello\ Lenzerini\}\}$ il programma deve restituire `NIL`.

Modelli e Complessità di Calcolo - Compito B

5 aprile 2005

Cognome:. . . . .
Nome:. . . . .
Matricola: . . . .

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire la Macchina di Turing deterministica (MT). Definire i concetti di linguaggio riconosciuto e linguaggio accettato da una MT. Quanto spazio di lavoro richiede una MT con un nastro di input unidirizionale e un nastro di lavoro per riconoscere il linguaggio $(a+b)^*b(a+b)^2$?

2. Definire i linguaggi decidibili e semidecidibili. Dimostrare che il linguaggio $\overline{K} = \big\{x \mid \varphi_x(x)$ non è definita$\big\}$ non è semidecidibile. Definire la Gerarchia Aritmetica. A che classe della Gerarchia Aritmetica appartiene $\overline{K}$?

3. Si considerino i seguenti problemi:
   Set Cover (SC): Dati un insieme S, una collezione $C = \{c_1, c_2, \ldots, c_m\}$ di sottoinsiemi di S e una costante k, decidere se esiste un sottoinsieme $C' \subseteq C$ tale che $|C'| \leq k$ e ogni elemento di S appartiene almeno a un sottoinsieme contenuto in C'.
   Insieme dei Delegati (ID): i dipendenti di una azienda sono divisi in classi rispetto a varie caratteristiche, ad esempio:

   - giovani (≤ 30 anni) e vecchi (> 30 anni);
   - uomini e donne;
   - impiegati e operai;
   - romanisti, laziali e non tifosi;
   - quelli che vengono al lavoro in auto, in moto, con i mezzi oppure a piedi;
   - ecc.

   Il problema consiste nel decidere se, data una classificazione dei dipendenti e un intero $k \geq 1$, è possibile costruire una delegazione di al più k dipendenti che partecipi a una riunione sindacale in modo che tra i k membri della delegazione , per ogni classe, ci sia almeno un componente che appartiene alla classe.

   Sapendo che il problema SC è NP-completo, mostrare che anche il problema ID è NP-completo.

4. Scrivere un programma LISP che, presa in ingresso una lista D contenente un insieme di dipendenti e una lista C contenente insiemi di dipendenti (raggruppati nelle rispettive classi di appartenenza), verifichi se D è una soluzione del problema ID descritto sopra. Ogni lista contenuta in C contiene come primo elemento la caratteristica della classe e poi i nomi dei dipendenti. Per esempio, se
   $C = \big\{\{Uomo\ Dilbert\ Wally\ Asok\ \}\ \{Donna\ Alice\ Carol\ Tina\ \}$
   $\{Consulente\ Dogbert\}\ \{Ingegnere\ Dilbert\}\{Patentato\ Dilbert\ Wally\ \}\big\}$
   e $D = \{\ Dilbert\ Dogbert\ Alice\}$ il programma deve restituire `T`;
   se $D = \{\ Dilbert\ Dogbert\ Wally\}$ il programma deve restituire `NIL`.

Modelli e Complessità di Calcolo - Compito A

13 aprile 2005

Cognome:. . . . .
Nome:. . . . .
Matricola: . . . .

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire la Macchina di Turing non deterministica (MTND). Mostrare come si può simulare una MTND con una MT deterministica. Quanto tempo richiede una MTND, con un nasto di input unidirezionale e un nastro di lavoro bidirezionale, per riconoscere il linguaggio $\{ww^R | w \in \{a, b\}^+\}$?

2. Definire le funzioni ricorsive primitive. Mostrare che la funzione:

$$f(x, y) = \begin{cases} x + y & \text{se } x = y \\ x \cdot y & \text{se } x \neq y \end{cases}$$

   è ricorsiva primitiva.

3. Si considerino i seguenti problemi:
   Set Packing (SP): Data una collezione C di insiemi finiti e una costante $k \leq |C|$, C contiene almeno k insiemi disgiunti?
   Clique (CL): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$ decidere se esiste una clicque $V' \subseteq V$, $|V'| \leq k$, tale che, per ogni $v_1, v_2 \in V'$ l'arco $e = (v_1, v_2)$ appartenga ad E.

   Sapendo che il problema CL è NP-completo, mostrare che anche il problema SP è NP-completo.

4. Scrivere (e commentare!) un programma LISP che, presa in ingresso una lista contenente un insieme di vertici `C` e una lista di archi `E`, rappresentati come coppie di vertici, verifica se `C` è una clique. Per esempio, se `C=(A B D)` e `E=((A B)(A D)(A C)(B D))` il programma deve restituire `T`; se `C=(A C D)` e `E=((A B)(A D)(A C)(B D))` il programma deve restituire `NIL`.

Modelli e Complessità di Calcolo - Compito B

13 aprile 2005

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire la Macchina di Turing (MT) deterministica a più nastri (MTM). Mostrare come si puo simulare una MTM con una MT. Quanto tempo richiedono una MT ad un nastro ed una MT a due nastri (di cui uno di sola lettura unidirezionale) per riconoscere il linguaggio delle parentesi ben bilanciate?

2. Mostrare che le funzioni ricorsive sono calcolabili con il Semplice Linguaggio Funzionale (SLF). Definire con SLF la funzione
$$f(x,n) = \prod_{i=1}^{n} F(x,i)$$
(assumendo di conoscere la definizione della funzione F).

3. Si considerino i seguenti problemi:
Clique (CL): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$ decidere se esiste una clicque $V' \subseteq V$, $|V'| \leq k$, tale che, per ogni $v_1, v_2 \in V'$ l'arco $e = (v_1, v_2)$ appartenga ad E.
Set Packing (SP): Data una collezione C di insiemi finiti e una costante $k \leq |C|$, C contiene almeno k insiemi disgiunti?

   Sapendo che il problema SP è NP-completo, mostrare che anche il problema CL è NP-completo.

4. Scrivere (e commentare!) un programma LISP che, presa in ingresso una lista `L` contenente insiemi (rappresentati tramite liste), verifica se `L` rappresenta un Set Packing. Per esempio, se `L=((A B)(C D)(E F))` il programma deve restituire `T`; se `L=((A B)(C E)(E F))` il programma deve restituire `NIL`.

Modelli e Complessità di Calcolo - Compito A

27 giugno 2005

Cognome:..
Nome:..
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Data una funzione ricorsiva h totale, dimostrare che l'insieme $\{i|\varphi_i = h\}$ è indecidibile. A quale livello della gerarchia di Kleene appartiene?
2. Come si dimostra che ogni linguaggio di tipo 0 può essere accettato da una macchina di Turing?
3. Si considerino i seguenti problemi:
   K-Path (KP): Dato un intero $K \geq 0$ e un grafo completo $G = (V, E)$, dove ad ogni arco e_i è associato un peso intero w_i, decidere se esiste un cammino (aciclico) in cui la somma dei pesi sugli archi sia maggiore o uguale a K.
   K-Path $s - t$ (KPst): Dato un intero $K \geq 0$, un grafo completo $G = (V, E)$, dove ad ogni arco e_i è associato un peso intero w_i, e due vertici $s, t \in V$, decidere se esiste un cammino (aciclico) tra s e t in cui la somma dei pesi sugli archi sia maggiore o uguale a K.

   Sapendo che il problema KPst è NP-completo, mostrare che anche il problema KP è NP-completo.
4. Scrivere un programma LISP che, presa in ingresso una matrice, restituisca `T` se la matrice è triangolare superiore, e `NIL` altrimenti. La matrice è rappresentata come lista di liste che contengono, ordinati, gli elementi delle riga; come esempio, la matrice

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{pmatrix}$$

   viene rappresentata come `{{1 2 3}{0 4 5}{0 0 6}}`.

Modelli e Complessità di Calcolo - Compito B

27 giugno 2005

Cognome:. . . . .
Nome:. . . . .
Matricola: . . . .

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Dimostrare che l'insieme $\{i|\varphi_i$ è crescente$\}$ è indecidibile. A quale livello della gerarchia di Kleene appartiene?

2. Come si dimostra che ogni linguaggio accettato da una macchina di Turing è di tipo 0?

3. Si considerino i seguenti problemi:
   K-Path su grafo non completo (KP-GNC): Dato un intero $K \geq 0$ e un grafo connesso non completo $G = (V, E)$ decidere se esiste un cammino (aciclico) di lunghezza maggiore o uguale a K (ovvero composto da almeno K archi).
   K-Path $s - t$ su grafo non completo (KPst-GNC): Dato un intero $K \geq 0$, un grafo connesso non completo $G = (V, E)$ e due vertici $s, t \in V$, decidere se esiste un cammino (aciclico) tra s e t di lunghezza maggiore o uguale a K (ovvero composto da almeno K archi).

   Sapendo che il problema KPst-GNC è NP-completo, mostrare che anche il problema KP-GNC è NP-completo.

4. Scrivere un programma LISP che, presa in ingresso una matrice, restituisca `T` se la matrice è triangolare inferiore, e `NIL` altrimenti. La matrice è rappresentata come lista di liste che contengono, ordinati, gli elementi delle riga; come esempio, la matrice

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 3 & 0 \\ 4 & 5 & 6 \end{pmatrix}$$

   viene rappresentata come `{{1 0 0}{2 3 0}{4 5 6}}`.

Informatica Teorica - Compito A

Settembre 2005

Cognome:..
Nome:..
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire la Macchina di Turing multinastro (MTM). Definire il concetto di accettazione e rifiuto di stringhe da parte di una MTM. Illustrare la simulazione di una MTM mediante una MT a un nastro.
2. Enunciare e dimostrare il Pumping Lemma per i linguaggi *context free*.
3. Dimostrare la NP-completezza del problema Vertex Cover.
4. Relativamente al problema PdN definito sopra, scrivere un programma LISP `PdN` che, ricevute in ingresso la lista di antipatie (rappresentato come lista di coppie di persone) e una lista contenente la composizione dei tavoli (ogni tavolo è rappresentato da una lista contenente le persone assegnae a quel tavolo), restituisca T se l'assegnazione di posti è valida, ovvero se non ci sono antipatie in nessun tavolo, e NIL altrimenti.

   Per esempio, se l'input è costituito dalla lista di antipatie `L1=((Valeria Vincenzo) (Vincenzo Antonella) (Antonella Alessandro) (Alessandro Valeria)(Vincenzo Luca))` e dalla lista di tavoli `L2=((Valeria Antonella Luca) (Vincenzo Alessandro))` il programma dovrebbe restituire T.

Modelli e Complessità di Calcolo - Compito B

12 settembre 2005

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire la Macchina di Turing non deterministica (MTND). Definire il concetto di accettazione e rifiuto di stringhe da parte di una MTND. Illustrare la simulazione di una MTND mediante una MT deterministica.

2. Dimostrare che l'insieme $R_x = \{z \mid \exists y \text{ tale che } \varphi_x(z) = y\}$ è ricorsivamente enumerabile.

3. Si considerino i seguenti problemi:
   Sudoku $n^2 \times n^2$: Data una griglia di dimensione $n^2 \times n^2$, divisa in n^2 blocchi (quadrati) di dimensione $n \times n$, in cui alcune delle celle compaiono numeri compresi tra 1 e n^2, decidere se è possibile riempire tutte le celle vuote (usando sempre numeri compresi tra 1 e n^2) rispettando la seguente regola: *ogni numero può apparire una sola volta in ogni riga, in ogni colonna e in ogni blocco.*
   Graph Coloring con Vincoli (GCV): dati $k \geq 0$ colori, un grafo $G = (V, E)$ in cui ad alcuni dei vertici è stato assegnato un colore (scelto tra i k), è possibile colorare i vertici rimanenti in maniera che ogni coppia di vertici adiacenti abbia colori distinti?

   Mostrare che Sudoku$\leq$GCV.

4. Scrivere un programma LISP `GKC` che, ricevute in ingresso la descrizione di un grafo (rappresentato come lista di archi) e una lista contenente la colorazione di vertici (rappresentata come lista di liste di vertici con lo stesso colore), restituisca T se la colorazione è una colorazione valida, ovvero se non esistono due vertici adiacenti dello stesso colore, e NIL altrimenti.

   Per esempio, se l'input è costituito dalla lista di archi `L1=((A B) (B C) (C D) (D A)(B E))` e dalla lista di colori `L2=((A C E) (B D))` il programma dovrebbe restituire T.

Esempio di istanza di problema Sudoku con $n = 3$

2	5			3		9		1
	1				4			
4		7				2		8
		5	2					
				9	8	1		
	4				3			
			3	6			7	2
	7							3
9		3				6		4

Modelli e Complessità di Calcolo - Compito A

24 marzo 2006

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Dimostrare che una macchina di Turing a più nastri può essere simultata da una macchina di Turing a un solo nastro. Quale è il costo della simulazione?

2. Sia $G = (N, A)$ un grafo orientato. Dimostrare che il problema di decidere se un nodo di G è raggiungibile da un altro nodo dato è risolubile in spazio $O(\log^2 |N|)$. Che conseguenza ha questo risultato sulle classi di complessità?

3. Si considerino i seguenti problemi:
   Cavalieri della Tavola Rotonda (CTR): Dato un elenco di cavalieri, con le rispettive relazioni di amicizia, esiste una disposizione dei posti alla tavola rotonda tale che ogni cavaliere sia seduto tra due suoi amici? Notare che l'amicizia è simmetrica: se Lancillotto è amico di Tristano, allora Tristano è amico di Lancillotto.
   Ciclo Hamiltoniano (HC): Dato un grafo $G = (V, E)$ esiste un ciclo che tocca tutti i vertici una ed una sola volta?

   Si richiede di:

   (a) mostrare due istanze del problema HC, in cui il grafo abbia almeno 6 nodi e almeno 9 archi: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema CTR è NP-completo, mostrare che anche il problema HC è NP-completo.

4. Scrivere un programma LISP `HC` che verifica se un ciclo sia valido, ovvero composto di archi appartenenti al grafo. In particolare, presa in ingresso una lista `L` contenente il ciclo (in cui il primo nodo compare anche in ultima posizione) e il grafo `G`, rappresentato come lista di liste di adiacenza, il programma deve restituire `T` se il ciclo è valido, e `NIL` altrimenti. Nella rappresentazione con liste di adiacenza per ogni nodo c'è una lista di adiacenza: il primo elemento della lista è il nodo stesso, il secondo elemento della lista è la lista dei nodi adiacenti ad esso. Esempio: con `L=(A B C D A)` e `G=( (A (B C D)) (B (A C)) (C (A B D)) (D (C A))` il programma deve restituire `T`.

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito B

24 marzo 2006

Cognome:...
Nome:...
Matricola:..

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Dimostrare che una macchina di Turing a un nastro, con alfabeto qualsiasi, puo' essere simulata da una macchina di Turing con un nastro semi-infinito e con un alfabeto costituito da un solo simbolo oltre al blank.

2. Mostrare come si simula una macchina di Turing non deterministica con una macchina di Turing deterministica. Quanto tempo richiede la simulazione? Che conseguenza ha questo risultato sulle classi di complessita'?

3. Si considerino i seguenti problemi:
   Ciclo Hamiltoniano (HC): Dato un grafo $G = (V, E)$ esiste un ciclo che tocca tutti i vertici una ed una sola volta?
   Cavalieri della Tavola Rotonda (CTR): Dato un elenco di cavalieri, con le rispettive relazioni di amicizia, esiste una disposizione dei posti alla tavola rotonda tale che ogni cavaliere sia seduto tra due suoi amici? Notare che l'amicizia è simmetrica: se Lancillotto è amico di Tristano, allora Tristano è amico di Lancillotto.

   Si richiede di:

   (a) mostrare due istanze del problema CTR, con almeno 5 cavalieri ($c_1 \dots c_5$) e almeno 7 relazioni di amicizia: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema HC è NP-completo, mostrare che anche il problema CTR è NP-completo.

4. Scrivere un programma LISP `CTR` che verifica se una disposizione dei posti alla tavola rotonda è valida. In particolare, presa in ingresso una lista `L` contenente la disposizione dei posti (in cui il primo cavaliere compare anche in ultima posizione) e la lista `A` delle amicizie tra cavalieri (rappresentate come liste), il programma deve restituire `T` se la disposizione è valida, e `NIL` altrimenti. Esempio: con
   `L=(Artù Lancillotto Tristano Artù)` e
   `A=((Artù Lancillotto)(Artù Tristano)(Tristano Lancillotto))` il programma deve restituire `T`.

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito A

20 aprile 2006

Cognome:...
Nome:...
Matricola:..**Corso di Laurea: Triennale** ☐ **Specialistica** ☐

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire gli insiemi ricorsivamente enumerabili. Dimostrare che l'insieme

$$T = \{x \mid \varphi_x \text{ è una funzione totale}\}$$

non è ricorsivamente enumerabile.
(N.B. Esistono diverse possibili dimostrazioni, fornirne almeno una.)

2. Enunciare il teorema di gerarchia spaziale. Che conseguenze ha sulle classi di complessità spaziale? Dire se esistono i linguaggi L_1 ed L_2 tali che

$$L_1 \in DSPACE(n^2 \log\log n), L_1 \notin DSPACE(n^2)$$

$$L_2 \in DSPACE(n^2 + n), L_2 \notin DSPACE(n^2 + \ logn)$$

3. Si considerino i seguenti problemi:
Hitting Set (HS): Dati un insieme $S = \{e_1, e_2, \ldots, e_n\}$, m sottoinsiemi $C_1, C_2, \ldots, C_m$ di S e una costante k, decidere se esiste un sottoinsieme $S' \subseteq S$ tale che $|S'| \leq k$ e S' contiene almeno un elemento di ciascun $C_i, i = 1, 2, \ldots, m$.
Vertex Cover (VC): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$, decidere se esiste un vertex cover $V' \subseteq V$, $|V'| \leq k$, tale che, per ogni arco $e = (v_1, v_2) \in E$, almeno uno tra v_1 e v_2 sia contenuto in V'.

Si richiede di:

(a) mostrare due istanze del problema HS, in cui $|S| \geq 6$ e $m \geq 6$: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

(b) sapendo che il problema VC è NP-completo, mostrare che anche il problema HS è NP-completo.

4. Scrivere e commentare un programma LISP `VC` che verifica se un insieme V' sia un vertex cover. In particolare, presa in ingresso una lista `L` contenente l'insieme e il grafo `G`, rappresentato come lista di liste di adiacenza, il programma deve restituire `T` se è un vertex cover, e `NIL` altrimenti. Nella rappresentazione con liste di adiacenza per ogni nodo c'è una lista di adiacenza: il primo elemento della lista è il nodo stesso, il secondo elemento della lista è la lista dei nodi adiacenti ad esso. Esempio: con `L=(C E)` e `G=( (A (B C D)) (B (A C)) (C (A B D)) (D (C A) (E (F)))` il programma deve restituire `NIL`.

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito B

20 aprile 2006

Cognome:..
Nome:..
Matricola:..**Corso di Laurea: Triennale** □ **Specialistica** □

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire gli insiemi ricorsivi e gli insiemi ricorsivamente enumerabili. Dimostrare che l'insieme

$$\bar{K} = \{x \mid \varphi_x(x) \text{ non è definita}\}$$

non è ricorsivamente enumerabile.

2. Enunciare il teorema di gerarchia temporale. Che conseguenze ha sulle classi di complessità temporale? Dire se esistono i linguaggi L_1 ed L_2 tali che

$$L_1 \in DTIME(n^2 \log\log n), L_1 \notin DTIME(n^2)$$

$$L_2 \in DTIME(n^2 + \log^2 n), L_2 \notin DTIME(n^2)$$

3. Si considerino i seguenti problemi:
Hitting Set (HS): Dati un insieme $S = \{e_1, e_2, \ldots, e_n\}$, m sottoinsiemi $C_1, C_2, \ldots, C_m$ di S e una costante k, decidere se esiste un sottoinsieme $S' \subseteq S$ tale che $|S'| \leq k$ e S' contiene almeno un elemento di ciascun $C_i, i = 1, 2, \ldots, m$.
Dominating Set (DS): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$ decidere se esiste un dominating set $V' \subseteq V$, $|V'| \leq k$, tale che ogni nodo $v_i \in V''$, $(V'' = V - V')$ sia collegato ad almeno un nodo in V' (ovvero esiste almeno un arco $e = (v_i, v_j) \in E$, tale che v_j sia contenuto in V').

   Si richiede di:

   (a) mostrare due istanze del problema DS, in cui $|V| \geq 6$ e $|E| \geq 9$: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema DS è NP-completo, mostrare che anche il problema HS è NP-completo.

4. Scrivere e commentare un programma LISP `DS` che verifica se un insieme V' sia un dominating set. In particolare, presa in ingresso una lista `L` contenente l'insieme e il grafo `G`, rappresentato come lista di liste di adiacenza, il programma deve restituire `T` se è un dominating set, e `NIL` altrimenti. Nella rappresentazione con liste di adiacenza per ogni nodo c'è una lista di adiacenza: il primo elemento della lista è il nodo stesso, il secondo elemento della lista è la lista dei nodi adiacenti ad esso. Esempio: con `L=(C E)` e `G=( (A (B C D)) (B (A C)) (C (A B D)) (D (C A) (E (F)))` il programma deve restituire `T`.

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Informatica Teorica - Compito A

VECCHIO ORDINAMENTO

20 aprile 2006

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Enunciare il pumping lemma per i linguaggi regolari. Dimostrare che i linguaggi $\{a^n b^n | n \geq 1\}$ e $\{ww^R | w \in \Sigma^+\}$ non sono regolari.

2. Enunciare il teorema di gerarchia spaziale. Che conseguenze ha sulle classi di complessità spaziale? Dire se esistono i linguaggi L_1 ed L_2 tali che

$$L_1 \in DSPACE(n^2 \log\log n), L_1 \notin DSPACE(n^2)$$

$$L_2 \in DSPACE(n^2 + n), L_2 \notin DSPACE(n^2 + \ logn)$$

3. Definire una grammatica per il linguaggio

$$\{w_1 w_1^R w_2 w_2^R \dots w_n w_n^R \mid w_i \in \Sigma^+,\ n \geq 1\}$$

.

4. Scrivere e commentare un programma LISP `VC` che verifica se un insieme V' sia un vertex cover. In particolare, presa in ingresso una lista `L` contenente l'insieme e il grafo `G`, rappresentato come lista di liste di adiacenza, il programma deve restituire `T` se è un vertex cover, e `NIL` altrimenti. Nella rappresentazione con liste di adiacenza per ogni nodo c'è una lista di adiacenza: il primo elemento della lista è il nodo stesso, il secondo elemento della lista è la lista dei nodi adiacenti ad esso. Esempio: con `L=(C E)` e `G=( (A (B C D)) (B (A C)) (C (A B D)) (D (C A) (E (F)))` il programma deve restituire `NIL`.

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Informatica Teorica - Compito B

VECCHIO ORDINAMENTO

20 aprile 2006

Cognome:...
Nome:...
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Enunciare il pumping lemma per i linguaggi context-free. Dimostrare che i linguaggi $\{a^n b^n c^n | n \geq 1\}$ e $\{ww | w \in \Sigma^+\}$ non sono context-free.

2. Enunciare il teorema di gerarchia temporale. Che conseguenze ha sulle classi di complessità temporale? Dire se esistono i linguaggi L_1 ed L_2 tali che

$$L_1 \in DTIME(n^2 \log\log n), L_1 \notin DTIME(n^2)$$

$$L_2 \in DTIME(n^2 + \log^2 n), L_2 \notin DTIME(n^2)$$

3. Definire una grammatica per il linguaggio

$$\{a^{n_1} b^{n_1} a^{n_2} b^{n_2} \dots a^{n_k} b^{n_k} \mid n_i \geq 1,\ k \geq 1\}$$

.

4. Scrivere e commentare un programma LISP `DS` che verifica se un insieme V' sia un dominating set. In particolare, presa in ingresso una lista `L` contenente l'insieme e il grafo `G`, rappresentato come lista di liste di adiacenza, il programma deve restituire `T` se è un dominating set, e `NIL` altrimenti. Nella rappresentazione con liste di adiacenza per ogni nodo c'è una lista di adiacenza: il primo elemento della lista è il nodo stesso, il secondo elemento della lista è la lista dei nodi adiacenti ad esso. Esempio: con `L=(C E)` e `G=( (A (B C D)) (B (A C)) (C (A B D)) (D (C A) (E (F)))` il programma deve restituire `T`.

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito A

5 luglio 2006

Cognome:..
Nome:..
Matricola:..

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire il concetto di 'enumerazione' delle funzioni ricorsive parziali (f.r.p.). Dimostrare che data una enumerazione delle f.r.p. esiste sempre un indice i tale che $\phi_i = \phi_{i^2}$.

2. Dimostrare l'indecidibilita' del problema della terminazione delle Macchine di Turing e illustrarne alcune conseguenze di interesse informatico.

3. Si considerino i seguenti problemi:
   Vertex Cover (VC): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$, decidere se esiste un vertex cover $V' \subseteq V$, $|V'| \leq k$, tale che, per ogni arco $e = (v_1, v_2) \in E$, almeno uno tra v_1 e v_2 sia contenuto in V'.
   Set Cover (SC): Dati un insieme S, una collezione $C = \{c_1, c_2, \ldots, c_m\}$ di sottoinsiemi di S e una costante k, esiste un sottoinsieme $C' \subseteq C$ tale che $|C'| \leq k$ e ogni elemento di S appartenga almeno a un sottoinsieme contenuto in C'?

   Si richiede di:

   (a) mostrare due istanze del problema SC, in cui l'insieme S abbia almeno 6 elementi e la collezione C abbia almeno 9 sottoinsiemi: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema VC è NP-completo, mostrare che anche il problema SC è NP-completo.

4. Scrivere e commentare un programma LISP `PC` che, ricevuta in ingresso una lista `L` contenente insiemi (rappresentati come liste), restituisca il prodotto cartesiano degli insiemi. Esempio: se `L=((A B)(1 2)(X Y))` il programma deve restituire `((A 1 X)(A 2 X)(B 1 X)(B 2 X)(A 1 Y)(A 2 Y)(B 1 Y)(B 2 Y))` (non necessariamente in quest'ordine).

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito B

5 luglio 2006

Cognome:..
Nome:..
Matricola: ....

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire il concetto di 'enumerazione' delle funzioni ricorsive parziali (f.r.p.). Dimostrare che data una enumerazione delle f.r.p. esiste sempre un indice i tale che, per ogni x, $\phi_i(x) = x^i$.

2. Dimostrare il Teorema di Savitch e discuterne le conseguenze sulle relazioni tra le classi di complessita' spaziali e temporali.

3. Si considerino i seguenti problemi:
   Dominating Set (DS): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$ decidere se esiste un dominating set $V' \subseteq V$, $|V'| \leq k$, tale che ogni nodo $v_i \in V''$, $(V'' = V - V')$ sia collegato ad almeno un nodo in V' (ovvero esiste almeno un arco $e = (v_i, v_j) \in E$, tale che v_j sia contenuto in V').
   Set Cover (SC): Dati un insieme S, una collezione $C = \{c_1, c_2, \ldots, c_m\}$ di sottoinsiemi di S e una costante k, esiste un sottoinsieme $C' \subseteq C$ tale che $|C'| \leq k$ e ogni elemento di S appartenga almeno a un sottoinsieme contenuto in C'?

   Si richiede di:

   (a) mostrare due istanze del problema DS, in cui il grafo abbia almeno 6 nodi e almeno 9 archi: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema DS è NP-completo, mostrare che anche il problema SC è NP-completo.

4. Scrivere e commentare un programma LISP `DIFSIM` che, ricevuta in ingresso una lista `L` contenente insiemi (rappresentati come liste), restituisca il differenza simmetrica degli insiemi. Si ricorda che la differenza simmetrica degli insiemi A e B è: $A \Delta B = A \cup B - A \cap B$. Esempio: se `L=((A B)(B C D)(D E F))` il programma deve restituire `(A C E F)` (non necessariamente in quest'ordine).

Il Sottoscritto, in base alla legge 675 del 31/12/96, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito A

20 aprile 2006

Cognome:..
Nome:..
Matricola:.. **Corso di Laurea: Triennale** ☐ **Specialistica** ☐

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire gli insiemi ricorsivi e gli insiemi ricorsivamente enumerabili. Dimostrare che un insieme ricorsivo è anche ricorsivamente enumerabile. Mostrare un esempio (con relativa dimostrazione) di un insieme ricorsivo e di un insieme ricorsivamente enumerabile ma non ricorsivo. Fornire una dimostrazione diretta (mediante diagonalizzazione) dei seguenti fatti:
   - L'insieme $T = \{x \mid f_x \text{ è una funzione totale}\}$ non è ricorsivo.
   - L'insieme $T = \{x \mid f_x \text{ è una funzione totale}\}$ non e' ricorsivamente enumerabile

2. Fornire una traccia della dimostrazione del teorema di Cook. Spiegare in quale (o in quali) parti della formula costruita nel teorema di Cook si vede che stiamo simulando macchine non deterministiche.

3. Si considerino i seguenti problemi:
   Massimo Uno per ogni Colonna (MUC): Data una matrice rettangolare A a elementi $0, 1$, con m righe e n colonne, e contenente *esattamente* due 1 per ogni colonna, e dato un intero $k > 0$, è possibile selezionare r righe, con $r > k$, in modo che ogni colonna della sottomatrice abbia al massimo un 1?
   Vertex Cover (VC): Dato un grafo $G = (V, E)$ e un intero $k \geq 0$, decidere se esiste un vertex cover $V' \subseteq V$, $|V'| \leq k$, tale che, per ogni arco $e = (v_1, v_2) \in E$, almeno uno tra v_1 e v_2 sia contenuto in V'.

   Si richiede di:

   (a) mostrare due istanze del problema MUC, in cui $m \geq 5$ e $n \geq 8$: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema VC è NP-completo, mostrare che anche il problema MUC è NP-completo.

4. Una lista di liste $L = \{l_1\ l_2\ l_3\ \dots\ l_n\}$, dove ogni l_i è una lista di interi, viene detta *onesta* se le sottoliste di cardinalità dispari contengono solo interi dispari e, similmente, se le sottoliste di cardinalità pari contengono solo interi pari. Per esempio, ((1 1 3) (2 2) (9)) è una lista *onesta*, mentre ((1 2) (3) (1 1)) non lo è. Scrivere e commentare un programma LISP che, presa in ingresso una lista di liste (di interi), restituisca `T` se la lista è onesta, e `NIL` altrimenti.

Il Sottoscritto, in base al d.l. 196 del 30/06/03, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito B

20 aprile 2006

Cognome:..
Nome:..
Matricola:...**Corso di Laurea: Triennale** ☐ **Specialistica** ☐

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire gli insiemi ricorsivi e gli insiemi ricorsivamente enumerabili. Dimostrare che un insieme ricorsivo è anche ricorsivamente enumerabile. Mostrare un esempio (con relativa dimostrazione) di un insieme ricorsivo e di un insieme ricorsivamente enumerabile ma non ricorsivo. Fornire una dimostrazione diretta (mediante diagonalizzazione) dei seguenti fatti:
   - L'insieme $T = \{x \mid f_x$ è una funzione totale$\}$ non è ricorsivo.
   - L'insieme $T = \{x \mid f_x$ è una funzione totale$\}$ non è ricorsivamente enumerabile

2. Dimostrare il teorema di Savitch e illustrare come è utilizzato nella dimostrazione che NP è contenuto in PSPACE.

3. Si considerino i seguenti problemi:
   Almeno Uno per ogni Colonna (AUC): Data una matrice rettangolare A a elementi $0, 1$, con m righe e n colonne, e contenente *esattamente* due 1 per ogni colonna, e dato un intero $k > 0$, è possibile selezionare r righe, con $r < k$, in modo che ogni colonna della sottomatrice abbia almeno un 1?
   Independent Set (IS): Dato un grafo $G = (V, E)$ e una costante $k \geq 0$ esiste un insieme indipendente $V' \subseteq V$, con $|V'| \geq k$, e tale che per ogni coppia di vertici $u, v \in V'$ l'arco $(u, v) \notin E$?

   Si richiede di:

   (a) mostrare due istanze del problema AUC, in cui $m \geq 5$ e $n \geq 8$: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema IS è NP-completo, mostrare che anche il problema AUC è NP-completo.

4. Una lista di liste $L = \{l_1\ l_2\ l_3\ \ldots\ l_n\}$, dove ogni l_i è una lista di interi, viene detta *crescente* se la serie costituita dalla somma degli elementi di ogni lista è una serie a valori crescenti. Per esempio, ((1 1) (1 2 3) (9)) è una lista *crescente*, mentre ((1 2) (4) (1 1)) non lo è. Scrivere e commentare un programma LISP che, presa in ingresso una lista di liste (di interi), restituisca `T` se la lista è crescente, e `NIL` altrimenti.

Il Sottoscritto, in base al d.l. 196 del 30/06/03, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito A

21 marzo 2007

Cognome: ....
Nome: ....
Matricola:....................................... **Corso di Laurea: Triennale** □ **Specialistica** □

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire gli insiemi semidecidibili e gli insiemi ricorsivamente enumerabili (r.e.) e dimostrare che sono la stessa classe di insiemi. Dimostrare che i linguaggi di tipo 0 sono semidecidibili e i linguaggi di tipo 1 sono decidibili.

2. Definire le classi di complessità P, NP ed EXPTIME e dimostrare le relazioni oggi note tra di esse. Supponiamo che si dimostri che il tempo necessario e sufficiente ad una macchina di Turing deterministica per simulare una macchina di Turing non deterministica operante in tempo $t(n)$ e' $t(n)^{\log t(n)}$. Che conseguenze ci sarebbero sulle relazioni tra le suddette classi?

3. Si considerino i seguenti problemi:
   Set Cover (SC): Dati un insieme S, una collezione $C = \{c_1, c_2, \ldots, c_m\}$ di sottoinsiemi di S e una costante k, esiste un sottoinsieme $C' \subseteq C$ tale che $|C'| \leq k$ e ogni elemento di S appartenga almeno a un sottoinsieme contenuto in C'?
   Hitting Set (HS): Dati un insieme $S = \{e_1, e_2, \ldots, e_n\}$, m sottoinsiemi $C_1, C_2, \ldots, C_m$ di S e una costante k, decidere se esiste un sottoinsieme $S' \subseteq S$ tale che $|S'| \leq k$ e S' contiene almeno un elemento di ciascun $C_i, i = 1, 2, \ldots, m$.

   Si richiede di:

   (a) mostrare due istanze del problema SC, in cui $|S| \geq 8$ e $m \geq 5$: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;

   (b) sapendo che il problema SC è NP-completo, mostrare che anche il problema HS è NP-completo.

4. Si consideri il seguente problema:
   Almeno Uno per ogni Colonna (AUC): Data una matrice rettangolare A a elementi $0, 1$, con m righe e n colonne, e contenente *esattamente* due 1 per ogni colonna, e dato un intero $k > 0$, è possibile selezionare r righe, con $r < k$, in modo che ogni colonna della sottomatrice abbia almeno un 1?

   Scrivere e commentare un programma LISP `AUC` che, ricevuta in ingresso una matrice (rappresentata come lista di liste, in cui ogni lista rappresenta una riga), verifichi se è una soluzione ammissibile del problema precedente, ovvero verifichi se in ogni colonna della matrice c'è almeno un 1. Ad esempio, se la matrice in ingresso è ((1 0 1) (0 1 1)) il programma deve restituire `T`.

Il Sottoscritto, in base al d.l. 196 del 30/06/03, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,

Modelli e Complessità di Calcolo - Compito B

21 marzo 2007

Cognome: ....
Nome: ....
Matricola:......................................**Corso di Laurea: Triennale** □ **Specialistica** □

(Si prega di compilare e riconsegnare assieme all'elaborato)

1. Definire i linguaggi semidecidibili. Dato un alfabeto Σ dimostrare che i linguaggi semidecidibili definiti su Σ^* sono di tipo 0 mostrando come si costruisce una grammatica di tipo 0 che genera L a partire dalla macchina M che accetta L. Considerare la seguente macchina di Turing che ha solo due regole di transizione:
   $d(q0, a) = (q0, a, d)$
   $d(q0, \not b) = (q1, \not b, i)$
   in cui lo stato $q0$ è lo stato iniziale, lo stato $q1$ è lo stato finale e $\Sigma = \{a, b\}$. Che linguaggio accetta tale macchina? Applicare la costruzione della grammatica di tipo 0 a tale macchina.

2. Definire le classi di complessità P, PSPACE ed EXPTIME e dimostrare le relazioni oggi note tra di esse. Supponiamo che si dimostri che i linguaggi accettati da una macchina di Turing deterministica in spazio $s(n)$ possono essere accettati da una macchina di Turing deterministica in tempo $s(n)^{\log s(n)}$. Che conseguenze ci sarebbero sulle relazioni tra le suddette classi?

3. Si considerino i seguenti problemi:
   Hitting Set (HS): Dati un insieme $S = \{e_1, e_2, \ldots, e_n\}$, m sottoinsiemi $C_1, C_2, \ldots, C_m$ di S e una costante k, decidere se esiste un sottoinsieme $S' \subseteq S$ tale che $|S'| \leq k$ e S' contiene almeno un elemento di ciascun $C_i, i = 1, 2, \ldots, m$.
   Set Cover (SC): Dati un insieme S, una collezione $C = \{c_1, c_2, \ldots, c_m\}$ di sottoinsiemi di S e una costante k, esiste un sottoinsieme $C' \subseteq C$ tale che $|C'| \leq k$ e ogni elemento di S appartenga almeno a un sottoinsieme contenuto in C'?

   Si richiede di:

   (a) mostrare due istanze del problema HS, in cui $n \geq 8$ e $m \geq 5$: una istanza che ammetta soluzione, con la relativa soluzione, e una istanza che non ammetta soluzione, argomentando l'assenza di soluzione;
   (b) sapendo che il problema HS è NP-completo, mostrare che anche il problema SC è NP-completo.

4. Si consideri il seguente problema:
   Massimo Uno per ogni Colonna (MUC): Data una matrice rettangolare A a elementi $0, 1$, con m righe e n colonne, e contenente *esattamente* due 1 per ogni colonna, e dato un intero $k > 0$, è possibile selezionare r righe, con $r > k$, in modo che ogni colonna della sottomatrice abbia al massimo un 1?

   Scrivere e commentare un programma LISP `MUC` che, ricevuta in ingresso una matrice (rappresentata come lista di liste, in cui ogni lista rappresenta una riga), verifichi se è una soluzione ammissibile del problema precedente, ovvero verifichi se in ogni colonna della matrice c'è al massimo un 1. Ad esempio, se la matrice in ingresso è ((1 0 1) (0 1 1)) il programma deve restituire `NIL`.

Il Sottoscritto, in base al d.l. 196 del 30/06/03, autorizza il Docente a pubblicare in bacheca e su Web i risultati della prova di esame. In fede,