Esercizi: la classe **NP**

1 Problemi

1.1 Problemi vari

Problema 9.1: Rispondere alla seguente domanda, dimostrando la propria affermazione: esiste un algoritmo polinomiale che trasforma una generica funzione booleana in forma disgiuntiva normale?

Problema 9.2: Il problema 2-COLORABILEOPPURE3-COLORABILE consiste nel chiedersi se un grafo G è 2-colorabile oppure è 3-colorabile.

Formalizzare la definizione del suddetto problema mediante la tripla $\langle I, S, \pi \rangle$, ed indicare la più piccola classe di complessità che lo contiene dimostrando la propria affermazione.

Problema 9.3: Si consideri il seguente problema decisionale NUMERIDIVISIBILI: dati due numeri interi positivi a e b, decidere se essi sono divisibili (ossia se uno dei due è multiplo dell'altro). In Tabella **??** è mostrato l'algoritmo **A** : **Div** che decide se una data coppia $\langle a, b \rangle$ di interi positivi appartiene a NUMERIDIVISIBILI. Dimostrare se **A** : **Div** è un algoritmo polinomiale o meno.

Problema 9.4: Sia $G = (V, E)$ un grafo non orientato e sia $D \subseteq V$; D è un insieme dominante per G se ogni nodo in $V - D$ ha un vicino in D. Il problema DOMINATING SET consiste nel decidere, dati un grafo $G = (V, E)$ e un intero k, se G contiene un insieme dominante di cardinalità $\leq k$.

1. Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I, S, \pi \rangle$, dimostrarne l'appartenenza alla classe **NP**.
2. La funzione f descritta di seguito è una riduzione polinomiale da VERTEX COVER a DOMINATING SET:

   sia $\langle G = (V, E), k \rangle$ una istanza di VERTEX COVER; allora $f(G, k) = \langle H = (W, F), k + 1 \rangle$ dove H è il grafo tale che

   - $W = V \cup X \cup \{a\}$, dove $X = \{x_e : e \in E\}$
     ed a è un nuovo nodo
   - $F = E \cup Y \cup Z$, con $Y = \{(u, x_e), (v, x_e) : e = (u, v) \in E\}$
     e $Z = \{(a, u) : u \in V$ è un nodo isolato$\}$.

Dimostrare che essa può essere calcolata in tempo polinomiale in $|G|$ e k e che se G ammette un Vertex Cover di al più k nodi allora H ammette un Dominating Set di al più $k+1$ nodi.

Suggerimento: il nodo a serve a gestire eventuali nodi isolati presenti nel grafo G.

Problema 9.5: Sia L il problema decisionale che consiste nel decidere, dati un insieme di interi $A = \{a_1, a_2, \ldots, a_n\} \subseteq \mathbb{N}$ e un intero $k \in \mathbb{N}$, se A contiene un sottoinsieme la somma dei cui elementi sia k. Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I, S, \pi \rangle$, dimostrarne l'appartenenza alla classe **NP**. Infine, progettare un algoritmo deterministico che decida tale problema calcolandone la complessità.
Suggerimento: un sottoinsieme di Z può essere rappresentato mediante una stringa binaria di n caratteri, che, a sua volta, può essere associata ad un numero intero compreso fra 0 e ....

Problema 9.6: Si consideri il seguente problema decisionale: dati una funzione booleana f in forma congiuntiva normale ed un intero k, decidere se f è soddisfacibile da una assegnazione di verità che assegna il valore vero ad esattamente k variabili.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I, S, \pi \rangle$, dimostrare che l'esistenza di un algoritmo polinomiale che lo decide implica l'esistenza di un algoritmo polinomiale che decide SAT. Cosa si può dedurre da ciò?

Problema 9.7: Siano L_1 ed L_2 due linguaggi tali che

a) $|L_1 - L_2| = \{a_1, a_2, a_3\}$,

b) $|L_2 - L_1| = \{b_1, b_2\}$,

c) $L_2 \in$ **NPC**.

Dimostrare che, in queste ipotesi, $L_1 \in$ **NPC**.

Problema 9.8: Siano L un linguaggio in **NPC** e $a \in L$.
Dimostrare che se **P**$\neq$**NP** allora $L - \{a\} \notin$ **P**.

Problema 9.9: Sia LARGE DOMINATING SET (in breve, LDS) il problema decisionale descritto al Problema 2. Dimostrare se la funzione $f : I_{2COL} \to I_{LDS}$ descritta di seguito è una riduzione polinomiale da 2-COLORABILITÀ a LARGE DOMINATING SET: dato un grafo non orientato $G = (V, E)$, istanza di 2-COLORABILITÀ, la corrispondente istanza di LARGE DOMINATING SET secondo f è il medesimo grafo $G = (V, E)$, ossia $f(G) = G$.

Problema 9.10: Si ricordi il problema k-COLORABILITÀ, per ogni costante $k \in \mathbb{N}$, e se ne formalizzi la definizione mediante la tripla $\langle I, S, \pi \rangle$.
Si consideri la seguente funzione f che trasforma un grafo (non orientato) $G = (V, E)$ in un nuovo grafo (non orientato) $f(G) = G' = (V', E')$ tale che

- $V' = V \cup \{x\}$, ove $x \notin V$ (ossia, V' è ottenuto aggiungendo un nuovo elemento a V).
- $E' = E \cup \{(u, x) : u \in V\}$ (ossia, E' contiene tutti gli archi in E e tutti gli archi che connettono x ad un elemento di V).

Verificare se f è una riduzione polinomiale da 3-COLORABILITÀ a 4-COLORABILITÀ dimostrando la propria affermazione.

Problema 9.11: Sia $k \in \mathbb{N}$ una costante fissata. Si formalizzi il problema k-COLORABILITÀ mediante la tripla $\langle I, S, \pi \rangle$. Successivamente, si consideri la seguente funzione f: dato un grafo $G = (V, E)$, $f(G)$ è un nuovo grafo $G' = (V', E')$

tale che $V' = V \cup \{a,b,c\}$ (con $a,b,c \notin V$) e $E' = E \cup \{(a,b),(b,c),(c,a)\}$. Verificare se f è una riduzione polinomiale da 4-COLORABILITÀ a 3-COLORABILITÀ, dimostrando la propria affermazione.

Problema 9.12: Si ricordi il problema decisionale DOMINATING SET: dati un grafo (non orientato) $G = (V,E)$ ed un intero k, decidere se esiste $D \subseteq V$ tale che $|D| \leq k$ e, per ogni $u \in V - D$, esiste $v \in D$ tale che $(u,v) \in E$.

Si consideri il problema decisionale EDGE COVER seguente: dati un grafo (non orientato) $G = (V,E)$ ed un intero k, decidere se esiste $E' \subseteq E$ tale che $|E'| \leq k$ e, per ogni $(u,v) \in E$, esiste $z \in V$ tale che $(u,z) \in E'$ oppure $(v,z) \in E'$. Dopo aver formalizzato la definizione dei suddetti problemi mediante la tripla $\langle I,S,\pi\rangle$, si consideri la trasformazione di istanze di EDGE COVER in istanze di DOMINATING SET mediante la funzione $f(G,k) = \langle \overline{G} = (E,\overline{E}),k\rangle$ in cui $(e_1,e_2) \in \overline{E}$ se e solo se e_1 ed e_2 hanno in G un estremo in comune. Si verifichi se detta trasformazione è una riduzione polinomiale da EDGE COVER a DOMINATING SET e si verifichi se essa è sufficiente a provare che EDGE COVER è **NP**-completo.

Problema 9.13: Si consideri il seguente problema: dati un grafo (non orientato) $G = (V,E)$ ed una coppia di nodi $u,v \in V$, decidere se esiste in G un percorso semplice (ossia, che non passi più volte per lo stesso nodo) da u a v di lunghezza (esattamente) $\left\lceil \frac{|V|}{2} \right\rceil - 1$.
Si consideri, ora, la seguente funzione f che trasforma istanze del problema CAMMINO HAMILTONIANO (HP) in istanze del problema in esame: data una istanza $G = (V,E)$ ed una coppia di nodi $u,v \in V$, $f(G,u,v) = \langle G' = (V',E),u,v\rangle$ in cui $V' = V \cup \overline{V}$ e $\overline{V}$ è un insieme di $|V|$ nodi, ossia, G' consiste del grafo $G = (V,E)$ e di ulteriori $|V|$ nodi isolati e u e v rimangono invariati.
Dimostrare se f è una riduzione polinomiale da HP al problema in esame, e se questo dimostra che il problema in esame è **NP**-completo.

Problema 9.14: Si ricordi il problema DOMINATING SET (DS): dati un grafo non orientato $G = (V,E)$ ed un intero $k \in \mathbb{N}$, decidere se esiste $D \subseteq V$ tale che $|D| \leq k$ e, per ogni $u \in V - D$, esiste $v \in D$ tale che $(u,v) \in E$.

Si consideri, ora, il seguente problema 2-HOPS-DOMINATING SET (2HDS): dati un grafo non orientato $G = (V,E)$ ed un intero $k \in \mathbb{N}$, decidere se esiste $H \subseteq V$ tale che $|H| \leq k$ e, per ogni $u \in V - H$, esistono $v \in H$ e $z \in V$ tali che $(u,z) \in E$ e $(z,v) \in E$.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, si dimostri se la funzione $f : I_{DS} \rightarrow I_{2HDS}$ di seguito descritta è una riduzione polinomiale da DS a 2HDS.
Sia $\langle G = (V,E),k\rangle$ una istanza di DS; allora $f(G,k) = \langle G' = (V',E'),k\rangle$, con

- $V' = V \cup E$;
- $E' = \{(x,y) : x \in V \wedge y \in E \wedge$ in G l'arco y è incidente sul nodo $x\}$, o, equivalentemente, $E' = \{(x,y) : x \in V \wedge y \in E \wedge \exists z \in V\ [\ y = (x,z)\ ]\ \}$.

Problema 9.15: Un grafo bipartito completo è una grafo non orientato il cui insieme di nodi è partizionato in due sottoinsiemi V_1 e V_2 e gli archi connettono ogni elemento di V_1 ad ogni elemento di V_2: formalmente, $G = (V,E)$ è bipartito completo se $V = V_1 \cup V_2$, con $V_1 \cap V_2 = \emptyset$, e

$$(u,v) \in E \Leftrightarrow u \in V_1 \wedge v \in V_2,$$

Si consideri il seguente problema SOTTOGRAFO BIPARTITO COMPLETO (in breve, *SBC*): dati un grafo non orientato $G = (V,E)$ ed un intero $k \in \mathbb{N}$, decidere se G contiene un sottografo bipartito completo di almeno $2k$ nodi.

Si ricordi la definizione del problema CLIQUE e si consideri la seguente funzione f che trasforma istanze di CLIQUE in istanze di *SBC*: data una istanza $\langle G = (V,E),k\rangle$ di CLIQUE, $f(G,k) = \langle \overline{G} = (\overline{V},\overline{E}),k\rangle$ dove

- $\overline{V} = \{u_1,u_2 : u \in V\}$ (ossia, ogni nodo in V è sostituito da una coppia di nodi in $\overline{V}$),

- $\overline{E} = \{(u_1, u_2) : u \in V\} \cup \{(u_1, v_2), (u_2, v_1) : (u, v) \in E\}$ (ossia, $\overline{E}$ contiene tutti gli archi che collegano coppie di nodi, u_1 e u_2, corrispondenti allo stesso nodo u in V, e inoltre, ogni arco $(u, v) \in E$ è sostituito in $\overline{E}$ dalla coppia di archi (u_1, v_2) e (u_2, v_1)).

Dopo aver formalizzato la definizione del problema *SBC* mediante la tripla $\langle I, S, \pi \rangle$, si dimostri se f è una riduzione polinomiale da CLIQUE a *SBC*.

Problema 9.16: Si consideri il problema seguente: dati un grafo non orientato $G = (V, E)$ ed un intero $k \in \mathbb{N}$, decidere se V può essere partizionato in k sottoinsiemi $V_1, \ldots, V_k$ ciascuno dei quali induce un sottografo completo in G.

Dopo aver formalizzato la definizione del suddetto problema Γ mediante la tripla $\langle I_\Gamma, S_\Gamma, \pi_\Gamma \rangle$, e dopo aver ricordato la definizione del problema COLORABILITÀ, si consideri la seguente funzione $f : I_{\text{COL}} \rightarrow I_\Gamma$: per ogni $\langle G = (V, e), k \rangle \in I_{\text{COL}}$,

$$f(G, k) = \langle \overline{G} = (V, \overline{E}), k \rangle,$$

dove $\overline{E} = \{(u, v) : u \in V \wedge v \in V \wedge (u, v) \notin E\}$.
Si dimostri che f è una riduzione polinomiale da COL a Γ.

Problema 9.17: Dato un grafo $G = (V, E)$, sia $\chi(G) = \langle M, P \rangle$ una sua codifica in cui M è la matrice di adiacenza di G e P è l'insieme di tutti i sottoinsiemi di V.

Si consideri il seguente problema decisionale: dati un grafo (non orientato) $G = (V, E)$ ed un intero k, esiste un sottoinsieme V' di V di cardinalità al più k e tale che, per ogni $u \in V - V'$, tutti i nodi adiacenti a u sono in V'?

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I, S, \pi \rangle$, descrivere un algoritmo che, prendendo in input $\chi(G)$ e k, decide se $\langle G, k \rangle$ è una istanza sì del problema in tempo polinomiale in $|\chi(G)|$ e $|k|$.

Rispondere, infine, alla seguente domanda: l'esistenza di tale algoritmo è sufficiente a dimostrare l'appartenenza alla classe **P** del problema?

Problema 9.18: Dato un grafo $G = (V, E)$, sia $\chi(G) = \langle M, P \rangle$ una sua codifica in cui M è la matrice di adiacenza di G e

$$P = \{\langle V_1, V_2, V_3 \rangle : V_1, V_2, V_3 \subseteq V \ \wedge \ V_1 \cup V_2 \cup V_3 = V \ \wedge \ V_1 \cap V_2 = \emptyset \ \wedge \ V_1 \cap V_3 = \emptyset \ \wedge \ V_2 \cap V_3 = \emptyset\}.$$

Si consideri il seguente problema decisionale: dato un grafo (non orientato) $G = (V, E)$, esiste una partizione $\langle V_1, V_2, V_3 \rangle$ di V tale che, per ogni $i = 1, 2, 3$ e per ogni $u, v \in V_i$, $(u, v) \notin E$?

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I, S, \pi \rangle$, descrivere un algoritmo che, prendendo in input $\chi(G)$, decide se $\langle G, k \rangle$ è una istanza sì del problema in tempo polinomiale in $|\chi(G)|$.

Rispondere, infine, alla seguente domanda: l'esistenza di tale algoritmo è sufficiente a dimostrare l'appartenenza alla classe **P** del problema?

Problema 9.19: Si consideri il problema seguente: dato un intero n, decidere se esistono due interi $h > 1$ e $k > 1$ tali che $n = hk$.

2.a) Dire se il seguente frammento di codice è un algoritmo non deterministico che decide il problema, argomentando la propria risposta:

1. **Input**: n;
2. **scegli** l'intero h nell'insieme $\{2, \ldots, n-1\}$;
3. **scegli** l'intero k nell'insieme $\{2, \ldots, n-1\}$;
4. **if** $(n = h \cdot k)$ **then Output**: accetta;
5. **else Output**: rigetta.

2.b) Dire se il seguente frammento di codice che decide il problema opera in tempo polinomiale, argomentando la propria risposta:

1. **Input**: n;
2. **for** $(h \leftarrow 2;\ h \leq n-1;\ h \leftarrow h+1)$ **do**
3.   **for** $(k \leftarrow 2;\ k \leq n-1;\ k \leftarrow k+1)$ **do**
4.   **if** $(n = h \cdot k)$ **then Output**: accetta;
5. **Output**: rigetta.

Anche alla luce dei punti 2.a) e 2.d) sopra, cosa si può dire circa l'appartenenza del problema alla classe **P** o alla classe **NP**? Motivare la propria risposta.

1.2 Appartenenza a NPC

Problema 9.20: Si consideri il seguente problema decisionale: dato un grafo non orientato $G = (V,E)$ (in cui V è l'insieme dei nodi ed E l'insieme degli archi) ed un intero positivo k, decidere se l'insieme V può essere partizionato in al più k insiemi indipendenti.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne la NP-completezza.

Problema 9.21: Si consideri il seguente problema decisionale: dato un grafo non orientato $G = (V,E)$ (in cui V è l'insieme dei nodi ed E l'insieme degli archi) ed un intero positivo k, decidere se l'insieme V può essere partizionato in al più k clique.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne la NP-completezza.

Problema 9.22: Il problema 4-SODDISFACIBILITÀ consiste nel chiedersi se una funzione booleana in forma congiuntiva normale con clausole di esattamente 4 letterali ciascuna è soddisfacibile.

Formalizzare la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, e dimostrarne la NP-completezza.

Problema 9.23: Si ricordi la definizione del problema CIRCUITO HAMILTONIANO.

Il problema PERCORSO HAMILTONIANO consiste nel chiedersi, dati un grafo non orientato $G = (V,E)$ ed una coppia di nodi $u, v \in V$, se esiste in G un percorso da u a v che tocchi tutti i nodi in V una ed una sola volta.

Il problema LONGEST PATH consiste nel chiedersi, dati un grafo non orientato $G = (V,E)$, una coppia di nodi $u, v \in V$ ed un intero positivo k, se esiste in G un percorso da u a v di lunghezza almeno k.

Dopo aver formalizzato la definizione del problema LONGEST PATH mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne l'appartenenza alla classe **NP** e la completezza rispetto a tale classe. In particolare, al fine di dimostrarne la completezza si richiede di mostrare una riduzione da CIRCUITO HAMILTONIAMO a PERCORSO HAMILTONIANO e poi un riduzione da PERCORSO HAMILTONIANO a LONGEST PATH.

Problema 9.24: Si ricordi la definizione di colorabilità di un grafo.

Dati un grafo $G = (V,E)$ e $V' \subseteq V$, il *grafo indotto* in G da V' è il grafo $G' = (V',E')$ in cui, per ogni coppia di nodi $x, y \in V'$, $(x,y) \in E'$ se e soltanto se $(x,y) \in E$.

Si consideri il seguente problema decisionale: dato un grafo non orientato $G = (V,E)$ ed un intero positivo k, decidere se l'insieme V contiene un sottoinsieme V' di almeno k nodi tale che il sottografo di G indotto da V' sia 1-colorabile.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.25: Sia +2SAT il problema decisionale seguente:

- $I_{+2SAT} = \{\langle f,k\rangle : f : X \to \{vero, falso\}$ è una funzione booleana in forma 2-congiuntiva normale non contenente alcun elemento di X negato e k è un intero positivo $\}$.
- $S_{+2SAT}(f,k) = \{a : X \to \{vero, falso\}\}$.
- $\pi_{+2SAT}(f,k,a) = f(a(X)) \wedge |\{x \in X : a(x) = vero\}| \leq k$.

Si dimostri la **NP**-completezza del suddetto problema.

Suggerimento: riduzione mediante VERTEX COVER.

Problema 9.26: Si consideri il seguente problema decisionale: dati un grafo (non orientato) $G = (V,E)$ ed un intero k, decidere se G soddisfa *almeno una* delle seguenti due proprietà:

a) G è un grafo di k nodi

b) esiste per G un Vertex Cover (ossia, un ricoprimento tramite nodi) di cardinalità al più k.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.27: Si consideri il seguente problema decisionale: dati un grafo (non orientato) $G = (V,E)$ ed un intero k, decidere se i nodi di G possono essere colorati con almeno k colori in accordo alle seguenti regole:

a) per ogni $i, j = 1,\ldots,k-1$ tali che $i \neq j$, se due nodi hanno colore, rispettivamente, i e j allora essi non sono collegati da un arco;

b) nessun vincolo è posto per nodi colorati con il colore k.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.28: Si consideri il seguente problema decisionale: dati un grafo (non orientato) $G = (V,E)$ ed un intero k, decidere se i nodi di G possono essere colorati con esattamente 2 colori in accordo alle seguenti regole:

a) ogni nodo colorato con il colore 1 è adiacente solo a nodi colorati con il colore 2;

b) il numero di nodi colorati con il colore 1 è almeno k.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.29: Si consideri il problema seguente: dato un grafo (non orientato) $G = (V,E)$, decidere se G è 3-colorabile oppure è un grafo completo.
Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.30: Si consideri il problema seguente: dati un grafo (non orientato) $G = (V,E)$ ed un intero $k \in \mathbb{N}$, decidere se G è 3-colorabile oppure contiene un sottografo completo di almeno $|V| - k$ nodi.
Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.31: Si consideri il problema seguente: dati un insieme X di variabili booleane, una funzione booleana in forma 3-congiuntiva normale f definita sull'insieme X, ed un intero $k \in \mathbb{N}$, decidere se esiste una assegnazione di verità agli elementi di X che soddisfa f e che assegna il valore `vero` ad al più k elementi di X.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, se ne dimostri l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.32: Si consideri il problema seguente: dati un insieme X di variabili booleane, una funzione booleana in forma 3-congiuntiva normale f definita sull'insieme X, ed un intero positivo $k \in \mathbb{N}^+$, decidere se esiste una assegnazione di verità agli elementi di X che soddisfa f e che assegna il valore `vero` ad almeno k elementi di X.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, se ne dimostri l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.33: Sia $k \in \mathbb{N}$ una costante positiva tale che $k \geq 3$.
Si consideri il problema seguente: dato un grafo non orientato $G=(V,E)$, decidere se G è k-colorabile oppure ha un Vertex Cover di cardinalità k.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, se ne dimostri l'appartenenza alla classe **P** o, in alternativa, la **NP**-completezza.

Problema 9.34: Si consideri il seguente problema decisionale: dati un grafo (non orientato) $G=(V,E)$ ed un intero $k \in \mathbb{N}$, con $k \geq 3$, decidere se (almeno) una delle seguenti due affermazioni è vera:

- G è 2-colorabile
- G contiene un sottografo completo di almeno k nodi.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, dimostrarne la **NP**-completezza o, in alternativa, l'appartenenza alla classe **P**.

1.3 coNP

Problema 9.35: Si consideri il seguente problema NO-CLIQUE: dato un grafo $G=(V,E)$ ed un intero $k>0$, decidere se ogni sottoinsieme di V di almeno k nodi contiene almeno una coppia di nodi non adiacenti.

Formalizzare la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, ed indicare la più piccola classe di complessità che lo contiene dimostrando la propria affermazione.

Problema 9.36: Si consideri il seguente problema decisionale: dato un grafo non orientato $G=(V,E)$ (in cui V è l'insieme dei nodi ed E l'insieme degli archi) ed un intero positivo k, decidere se ogni sottoinsieme di V di almeno k nodi contiene almeno una coppia di nodi adiacenti.

Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I,S,\pi\rangle$, collocarlo nella corretta classe di complessità.

Problema 9.37: Si consideri il problema decisionale seguente: dati un grafo non orientato $G=(V,E)$ e un intero positivo k, decidere se, comunque si scelga un nodo $u \in V$, non esistono k nodi $u_1,u_2,\ldots,u_k \in V$ tali che $\langle u,u_1,\ldots,u_k\rangle$ è una clique in G.

Collocare il suddetto problema nella corretta classe di complessità.

Problema 9.38: Si consideri il problema decisionale seguente: dato un grafo non orientato $G=(V,E)$, decidere se ogni ciclo in G ha lunghezza $<|V|-1$. Studiare la complessità computazionale di tale problema.
Si ricordi che un ciclo di lunghezza k in un grafo non orientato $G=(V,E)$ è una sequenza $\langle v_1,v_2,\ldots v_k\rangle$ di elementi distinti di V tali che, per $i=1,\ldots k-1$, $(v_i,v_{i+1}) \in E$ e, inoltre, $(v_k,v_1) \in E$.

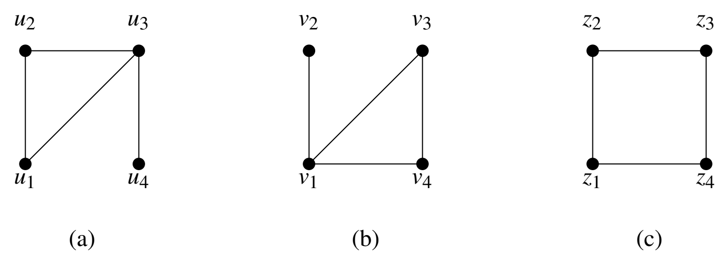

Figura 9.1: I grafi (a) e (b) sono isomorfi, ma non il grafo (c).

Problema 9.39: Si consideri il seguente problema decisionale: dato un grafo $G = (V, E)$ ed un intero $k \in \mathbb{N}$, decidere se, per ogni coppia di nodi $(u, v) \in V$, il percorso più lungo fra u e v ha lunghezza $\leq k$. Si studi la complessità computazionale di tale problema.

Problema 9.40: Due grafi $G = (V, E)$ e $G' = (V', E')$ sono *isomorfi* se essi sono uguali a meno di una ridenominazione dei nodi di uno dei due. Si vedano, ad esempio, i tre grafi riportati in Figura 9.1.

Più formalmente, G e G' sono isomorfi se esiste una biezione $b : V \to V'$ tale che, per ogni coppia di nodi $u, v \in V$ di G, si ha che $(u, v) \in E$ se e soltanto se $(b(u), b(v)) \in E'$. In riferimento alla figura, l'isomorfismo fra i grafi (a) e (b) è individuato dalla seguente biezione: $b(u_1) = v_3$, $b(u_2) = v_4$, $b(u_3) = v_1$, $b(u_4) = v_2$.

Il problema GRAFI NON ISOMORFI GNI) è definito nel modo seguente: dati due grafi (non orientati) $G = (V, E)$ e $G' = (V', E')$, decidere se G e G' non sono isomorfi.
Dopo aver formalizzato la definizione del suddetto problema mediante la tripla $\langle I, S, \pi \rangle$, dimostrarne l'appartenenza alla classe **P** o alla classe **NP** o alla classe co**NP**.

Problema 9.41: Si considerino i due problemi seguenti:

a) dati un grafo non orientato $G = (V, E)$ ed un intero $k \in \mathbb{N}$, decidere se G ha un Vertex Cover di cardinalità $> k$;

b) dati un grafo non orientato $G = (V, E)$ ed un intero $k \in \mathbb{N}$, decidere se ogni Vertex Cover in G ha cardinalità $> k$.

Dopo aver formalizzato la definizione dei suddetti problemi mediante la tripla $\langle I, S, \pi \rangle$, si collochi ciascuno di essi nella corretta classe di complessità.

Problema 9.42: Si considerino i due problemi seguenti:

a) dati un grafo non orientato $G = (V, E)$ ed un intero $k \in \mathbb{N}$, decidere se G contiene una Clique di cardinalità $< k$;

b) dati un grafo non orientato $G = (V, E)$ ed un intero $k \in \mathbb{N}$, decidere se ogni Clique in G ha cardinalità $< k$.

Dopo aver formalizzato la definizione dei suddetti problemi mediante la tripla $\langle I, S, \pi \rangle$, si collochi ciascuno di essi nella corretta classe di complessità.

2 Soluzioni

2.1 Problemi vari

Soluzione del problema 9.1

Supponiamo per assurdo che esista un algoritmo polinomiale **A:TODNF** che, data una funzione booleana f, calcola una funzione booleana g in forma disgiuntiva normale tale che f è soddisfacibile se e soltanto se g è soddisfacibile. Allora, possiamo decidere se una funzione booleana (ad esempio, in forma congiuntiva normale) è soddisfacibile mediante il seguente algoritmo:

A:RGH
input: funzione booleana $f(x_1, \dots, x_n)$;
output: accetta o rigetta;
  $g \leftarrow$ **A:TODNF**(f);
  if (**A:DNFSAT**(g) accetta) **output**: accetta;
  else output: rigetta.

Poiché per ipotesi l'algoritmo **A:TODNF** opera in tempo polinomiale, e poiché anche l'algorimo **A:DNFSAT** opera in tempo polinomiale (si veda il Problema ???), l'algoritmo **A:RGH** decide il problema SODDISFACIBILITÀ in tempo polinomiale. Ma, poiché SODDISFACIBILITÀ è un problema NP-completo, questo dimostrerebbe P= NP, da cui l'assurdo.

Soluzione del problema 9.2

Osserviamo subito che ogni 2-colorazione di un qualunque grafo è *anche* una 3-colorazione per esso che non usa uno dei tre colori consentiti. Pertanto,

- se un grafo G è 2-colorabile oppure 3-colorabile allora è anche 3-colorabile,
- se un grafo G è 3-colorabile allora è anche 2-colorabile oppure 3-colorabile,

ossia, il problema 2-COLORABILEOPPURE3-COLORABILE coincide con il problema 3-COLORABILITÀ ed è NP-completo. Dunque, esso può essere formalizzato come segue:

- $I = \{G = (V,E) : G$ è un grafo $\}$;
- $S = \{< c : V \rightarrow \{1,2,3\}\}$;
- $\pi(G,c) = c$ è una effettiva 3-colorazione dei nodi di G (ossia, per ogni $u,v \in V$, $c(u) = c(v) \Rightarrow (u,v) \notin E$).

Soluzione del problema 9.3

Senza perdita di generalità, assumiamo che $a \geq b$. Allora, la complessità dell'algoritmo **A** : **Div** in Tabella **??** è $\mathbf{O}(\lceil \frac{a}{b} \rceil)$. In particolare, il caso peggiore per l'algoritmo **A** : **Div** si verifica quando $b = 1$. In questo caso, il tempo necessario all'algoritmo a calcolare il proprio output è $\mathbf{O}(a)$.
Osserviamo, ora, che la dimensione dell'input è $|a| + |b| = \log a + \log b$ che, nel caso $b = 1$, diventa $|a| = \log a$. Questo è sufficiente a concludere che l'algoritmo **A** : **Div** ha complessità esponenziale nella *dimensione* dell'istanza.

Soluzione del problema 9.4

Il problema DOMINATING SET può essere formalizzato nella maniera seguente:

- $I_{DS} = \{\langle G = (V,E),k\rangle :\ G$ è un grafo non orientato completo e k un intero positivo $\}$.
- $S_{DS}(G,k) = \{D \subseteq V\}$.
- $\pi_{DS}(G,k,D) = |D| \leq k \ \wedge\ \forall u \in V - D[\exists v \in D : (u,v) \in E]$.

Il problema è in **NP**: infatti, un sottoinsieme dell'insieme V dei nodi può essere generata non deterministicamente in tempo O(n). Allo scopo, indichiamo con $V = \{v_1, v_2, \ldots, v_n\}$ l'insieme dei nodi di G e consideriamo un albero delle computazioni di profondità n e grado di non determinismo 2 in cui un nodo a livello $j = 1,2,\ldots,n-1$ corrisponde ad un sottoinsieme X di $\{v_1, v_2, \ldots v_j\}$: i due figli di tale nodo corrispondono ai due sottoinsiemi $X \cup \{v_{j+1}\}$ e X di $\{v_1, \ldots v_{j+1}\}$. In tal modo, le foglie dell'albero corrispondono ai sottoinsiemi di V. Una volta generato un sottoinsieme V' di V, come appena descritto, è sufficiente verificare se $|V'| \leq k$ (in tempo deterministico in $O(\min(n,k))$) e se ogni nodo in $V - V'$ è adiacente a qualche nodo in V' (in tempo deterministico in $O(kn|E|)$).

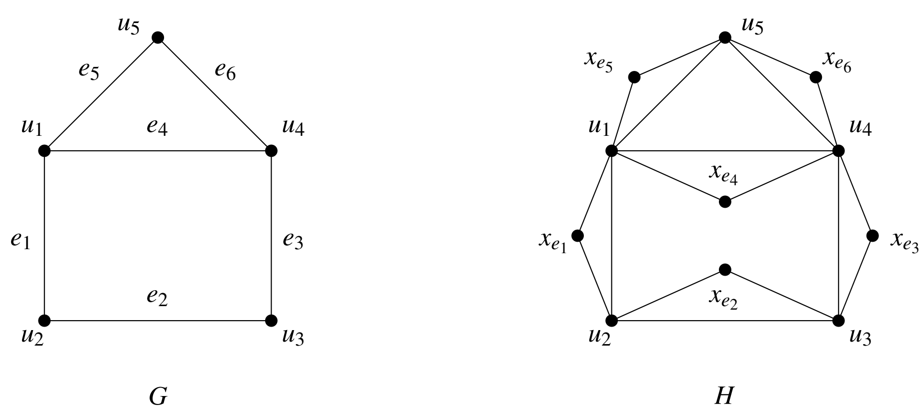

Figura 9.2: Il grafo G, istanza di VERTEX COVER ed il suo corrispondente H tramite la funzione f. In G sono stati anche evidenziati i nomi degli archi: all'arco e_i di G corrisponde il nodo x_{e_i} di H, per $i = 1,\ldots,6$.

Mostriamo ora che, dato un grafo non orientato G ed un intero positivo k, $f(G,k) = \langle H, k+1\rangle$ è calcolabile in tempo polinomiale in $|G|$ e k. Infatti, detto $G = (V,E)$, l'insieme dei nodi del grafo H è costruito aggiungendo all'insieme V un nodo per ogni arco in E, mentre l'insieme degli archi di H è costruito aggiungendo all'insieme E una coppia di archi per ogni arco in E:

1) $X \leftarrow \emptyset$;
2) **for** $(e \in E)$ **do** $X \leftarrow X \cup \{x_e\}$;
3) $W \leftarrow V \cup X \cup \{a\}$;
4) $Y \leftarrow \emptyset$;
5) **for** $(e = (u,v) \in E)$ **do** $Y \leftarrow Y \cup \{(u,x_e),(v,x_e)\}$;
6) $Z \leftarrow \emptyset$;
7) **for** ($u \in V$ isolato in G) **do** $Z \leftarrow Z \cup \{(a,u)\}$;
8) $F \leftarrow E \cup Y \cup Z$;
9) **Output**: $H = (W,F)$.

Questo richiede tempo in $O(|E|+|V|)$.

Resta da far vedere che se $\langle G=(V,E),k\rangle$ è una istanza sì di VERTEX COVER allora $f(G,k)=\langle H=(V\cup X,E\cup Y),k+1\rangle$ è una istanza sì di DOMINATING SET. A questo scopo, dato un insieme dominante C per un grafo, diciamo che un nodo b non contenuto in C è *dominato* da un nodo $c\in C$ se (b,c) è un arco del grafo.
Supponiamo, allora, che $G=(V,E)$ ammetta un Vertex Cover di k nodi: in tal caso, esiste $V'\subseteq V$ tale che $|V'|\leq k$ e, per ogni arco $(u,v)\in E$, $u\in V'$ oppure $v\in V'$. Proviamo, ora, che $D=V'\cup\{a\}$ è un Dominating Set per H. Infatti, per ogni $u\in W-D=W-(V'\cup\{a\})$, sono possibili i soli tre casi seguenti: $u\in V$ ed è adiacente a qualche nodo $v\in V$, oppure $u\in V$ ed è isolato in G, o, infine, $u\in X$. Di seguito, analizziamo separatamente i tre casi.

- $u\in V$ ed è adiacente a qualche nodo $v\in V$ (ossia, $(u,v)\in E$). In questo caso, poiché V' è un vertex cover per G e poiché $u\notin V'$, allora $v\in V'\subset D$ e, quindi, u è dominato da v.
- $u\in V$ ed è isolato in G. In questo caso, u è dominato da $a\in D$.
- $u\in X$. In questo caso, $u=x_e$, per qualche $e=(v_1,v_2)\in E$, e quindi, poiché $v_1\in V'\vee v_2\in V'$ e $(x_e,v_1)\in F\wedge(x_e,v_2)\in F$, $u=x_e$ è dominato da v_1 o da v_2.

Dunque, D è un insieme dominante per G.

Soluzione del problema 9.5

Problema 2. Il problema L può essere formalizzato nella maniera seguente:

- $I_L=\{\langle A,k\rangle:\ A=\{a_1,\ldots,a_n\}\subset\mathbb{N}$ e k è un intero positivo $\}$.
- $S_L(A,k)=\{A'\subseteq A\}$.
- $\pi_L(A,k,A')=\sum_{a\in A'}a\leq k$.

Per provare che il problema è in **NP** occorre progettare un algoritmo non deterministico che lo decide in tempo polinomiale. Tale algoritmo è rappresentabile mediante un albero delle computazioni binario e di altezza n: i nodi al livello i dell'albero (con $1\leq i\leq n$) corrispondono a tutti i sottoinsiemi possibili dell'insieme $\{a_1,a_2,\ldots,a_i\}$. Esso è descritto dal seguente pseudo-codice:

Input: $A=\{a_1,\ldots,a_n\}\subset\mathbb{N}$, $k\in\mathbb{N}$.

$A'\leftarrow\emptyset$;

for $(i\leftarrow 1;\ i\leq n;\ i\leftarrow i+1)$ **do**

  scegli se inserire a_i in A'; *(passo non deterministico)*

if $\sum_{a\in A'}a\leq k$ **then Output:** accetta;

else Output: rigetta.

Affrontiamo ora la progettazione di un algoritmo deterministico che decida L. Come indicato nel suggerimento, fissato un insieme $A=\{a_1,a_2,\ldots a_n\}$, i suoi sottoinsiemi sono in corrispondenza biunivoca con l'insieme delle stringhe binarie di lunghezza n: la stringa $b=b_1b_2\ldots b_n$ corrisponde al sottoinsieme di A

$$A_b=\{a_i\in A: b_i=1, i=1,\ldots,n\}.$$

Ma una stringa binaria di lunghezza n rappresenta un intero compreso fra 0 e 2^n-1: la stringa $b=b_1b_2\ldots b_n$ corrisponde all'intero $n_b=\sum_{i=1}^{n}b_i\cdot 10^{i-1}$.
L'algoritmo deterministico che decide L basato sulle precedenti osservazioni è descritto in Tavola 9.1: esso è costituito da un ciclo **while** che conta gli interi da 0 a 2^n-1 e verifica se l'insieme corrispondente ha peso non superiore a k

Input:	$A = \{a_1, a_2, \dots, a_n\} \subseteq \mathbb{N}$, $k \in \mathbb{N}$
1.	$sum \leftarrow 0$;
2.	$cont \leftarrow 0$;
3.	**while** ($cont \leq 2^n - 1 \wedge sum \neq k$) **do begin**
4.	$q \leftarrow cont$;
5.	$i \leftarrow 1$;
6.	**while** ($q > 0$) **do begin**
7.	$bin[i] \leftarrow q \ mod \ 2$;
8.	$q \leftarrow q/2$;
9.	$i \leftarrow i+1$;
10.	**end**
11.	$sum \leftarrow 0$;
12.	**for** ($i \leftarrow 1$; $i \leq n$; $i \leftarrow i+1$) **do**
13.	**if** ($bin[i] = 1$) **then** $sum \leftarrow sum + a_i$;
14.	$cont \leftarrow cont + 1$;
15.	**end**
16.	**if** ($sum = k$) **then Output**: esiste;
17.	**else Output**: non esiste.

Tabella 9.1: Algoritmo che verifica se A contiene un sottoinsieme di peso k.

(linee 3-15). In particolare, durante un'iterazione del ciclo viene generata la stringa binaria corrispondente all'intero in forma di array (ciclo **while** alle linee 6-10) e viene calcolato il peso dell'insieme corrispondente (linee 12-13).
Il ciclo **while** esterno viene ripetuto 2^n volte, il costo di ciascuna iterazione è dominato dal ciclo **while** interno ($O(n)$) e dal ciclo **for** ($O(\log \sum_{i=1}^{n} a_i)$). Pertanto, il costo dell'algoritmo è $O(n2^n \log \sum_{i=1}^{n} a_i)$.

Soluzione del problema 9.6

Il problema, che denoteremo, in breve, BSAT (*Bounded* SAT), può essere formalizzato nella maniera seguente:

- $I_{BSAT} = \{\langle X, f, k\rangle : f$ è una funzione booleana in forma congintiva normale nelle variabili in X costituita da m clausole $\wedge\ k \in \mathbb{N}\}$.
- $S_{BSAT}(X, f) = \{a : X \rightarrow \{vero, falso\}\}$.
- $\pi_{BSAT}(X, f, a) = f(a(X)) \wedge [\ |\{x \in X : a(x) = vero\}| = k]$.

Supponiamo che esista un algoritmo deterministico *bSAT* che, con input f, X, e k, decide in tempo polinomiale in $|f|$, $|X|$ e $|k|$ se f è soddisfacibile da una assegnazione di verità che assegna il valore vero ad esattamente k variabili. Allora, potremmo decidere SAT con l'algoritmo (deterministico) in Tabella 9.2.
Indichiamo con $c(m,n,k)$ il polinomio che, in accordo con la nostra ipotesi, limita il costo dell'algoritmo *bSAT* quando il suo input consiste di una funzione costituita da m clausole su un insieme di n variabili booleane e di un intero k; allora, il costo dell'algoritmo in Tabella 9.2 è

$$\sum_{i=0}^{n} c(m,n,i) \leq (n+1) \max\{c(m,n,i) : 0 \leq i \leq n\}.$$

Quindi, anche l'algoritmo in Tabella 9.2 che decide SAT avrebbe costo polinomiale nella dimensione dell'input. Ma, poiché SAT è un problema **NP**-completo, questo è possibile solo se **P**=**NP**.
Allora, se $\mathbf{P} \neq \mathbf{NP}$, BSAT$\not\in$**P**.

Soluzione del problema 9.7

Per dimostrare che L_1 è **NP**-completo è necessario dimostrare che $L_1 \in \mathbf{NP}$ e che $L1$ è completo per **NP**.
Per ciò che riguarda l'appartenenza ad **NP**, poiché non abbiamo informazioni sulla struttura di L_1 non abbiamo la

Input:	X, $f = c_1 \wedge c_2 \wedge c_m$ con $c_j = \{l_{j_1}, \ldots, l_{j_{h_j}}\}$, per ogni $j = 1, \ldots, m$.
	$sat \leftarrow falso$;
	$k \leftarrow 0$;
	while $(k \leq n \ \wedge \ sat = falso)$ **do begin**
	if ($bSAT(f, X, k)$ accetta) **then** $sat \leftarrow vero$;
	else $k \leftarrow k + 1$;
	if $(sat = vero)$ **then Output**: accetta;
	else Output: rigetta.

Tabella 9.2: Algoritmo che decide SAT.

possibilità di progettare per esso un algoritmo non deterministico polinomiale. Ricorriamo, allora, ad una riduzione, ossia, dimostriamo l'appartenenza di L_1 ad **NP** mediante una riduzione da L_1 ad L_2. Assumiamo che $L_1, L_2 \subseteq \Sigma^*$ e consideriamo la seguente funzione

$$\forall x \in \Sigma^*, \ f(x) \ = \begin{cases} x & \text{se } x \notin \{a_1, a_2, a_3, b_1, b_2\} \\ a_1 & \text{se } x \in \{b_1, b_2\} \\ b_1 & \text{se } x \in \{a_1, a_2, a_3\}. \end{cases}$$

Il calcolo di f richiede tempo costante, quindi $f \in$ **FP**. Inoltre, banalmente, $x \in L_1$ se e soltanto se $f(x) \in L_2$. Infatti:

- se $x \in L_1$, allora o $x \in \{a_1, a_2, a_3\}$ e $f(x) = b_1 \in L_2$, oppure $x \in L_1 - \{a_1, a_2, a_3\} \subset L_2$ e $f(x) = x \in L_2$;
- se $x \notin L_1$, allora o $x \in \{b_1, b_2\}$ e $f(x) = a_1 \notin L_2$, oppure $x \in \Sigma^* - (L_1 \cup \{b_1, b_2\}) \subset \Sigma^* - L_2$ e $f(x) = x \notin L_2$.

Questo prova che $L_1 \leq L_2$ e, quindi, $L_1 \in$ **NP**.

Per mostrare la completezza di L_1 rispetto ad **NP** dobbiamo dimostrare l'esistenza della riduzione inversa, ossia, dobbiamo mostrare che $L_2 \leq L_1$. In effetti, poiché L_1 e L_2 sono linguaggi sullo stesso alfabeto Σ, a questo scopo è sufficiente osservare che la stessa funzione f definita sopra è anche una riduzione da L_2 a L_1: infatti, in maniera analoga a quanto dimostrato sopra, possiamo mostrare che $x \in L_2$ se e soltanto se $f(x) \in L_1$. Questo prova che $L_2 \leq L_1$ e, quindi, che L_1 è completo per **NP**.

Soluzione del problema 9.8

Supponiamo, per assurdo, che $L' = L - \{a\} \in$ **P**: allora esistono una macchina di Turing deterministica T' ed un intero k tali che, per ogni $y \in \Sigma^*$, $T'(y)$ termina in tempo $O(|y|^k)$ ed inoltre $T'(y)$ accetta se e soltanto se $y \in L'$.
Possiamo, allora, utilizzare T' per definire una macchina di Turing deterministica che decide L, il cui comportamento è illustrato dal seguente algoritmo:

input: $x \in \Sigma_1^*$.

1) se $x = a$ allora T accetta;
2) altrimenti se $T'(x) = q_A$ allora T: accetta;
3) altrimenti T rigetta.

Poiché il passo 2) richiede tempo in $O(|x|^k)$, tale algoritmo richiede tempo polinomiale in $|x|$, e questo prova che il problema L appartiene alla classe **P**. MA L è **NP**-completo, dunque deve essere **P**=**NP**, contraddicendo l'ipotesi **P**$\neq$**NP**.

Soluzione del problema 9.9

Si ricordi che, affinché f sia una riduzione polinomiale da 2COL a LDS, è necessario che f sia caolcolabile in tempo polinomiale e che, per ogni grafo $G \in I_{2COL}$, si abbia che G è istanza sì per 2COL se e soltanto se $f(G) = G$ è istanza sì per LDS. La funzione f è ovviamente calcolabile in tempo polinomiale (essendo la funzione identità); tuttavia, essa non è una riduzione polinomiale da 2-COL a LDS. Per dimostrare tale affermazione è sufficiente mostrare un grafo che non sia 2 colorabile ma ammetta un insieme dominante contenente al più la metà dei suoi nodi oppure un grafo che sia 2 colorabile ma non ammetta un insieme dominante contenente al più la metà dei suoi nodi. In quanto segue descriviamo due grafi che rispettano, ciascuno, una delle due proprietà appena enunciate.

a) Sia K_3 il grafo completo di 3 nodi: K_3 non è 2 colorabile ma ammette un insieme dominante costituito da 1 nodo.

b) Sia I_n il grafo costituito da n nodi isolati: I_n è 2 colorabile (in effetti è sufficiente un solo colore per colorare i suoi nodi) ma l'unico insieme dominante per I_n contiene tutti i suoi nodi.

Dunque, f non è una riduzione polinomiale da 2COL a LDS.

Soluzione del problema 9.10

Ricordiamo che il problema k-COLORABILITÀ consiste nel decidere se i nodi di un dato grafo possono essere colorati con k colori in modo tale che gli estremi di nessun arco siano colorati con lo stesso colore. Formalmente,

- $I_{k-col} = \{G = (V,E) : G \text{ è un grafo non orientato}\}$;
- $S_{k-col}(G) = \{c : V \to \{1,2,\ldots,k\}\ \}$;
- $\pi_{k-col}(G,c) = \forall (u,v) \in E[c(u) \neq c(v)]$.

Consideriamo, ora, la funzione f della traccia. Dato un grafo non orientato $G = (V,E)$, calcolare $G' = f(G)$ richiede tempo lineare in $|V|$. Pertanto, $f \in \mathbf{FP}$.
Per dimostrare che f è una riduzione da 3-COLORABILITÀ a 4-COLORABILITÀ resta da far vedere che un grafo $G = (V,E)$ è 3-colorabile se e soltanto se il corrispondente $G' = f(G)$ è 4-colorabile.
Supponiamo, allora che G sia 3-colorabile, ossia, che esista una funzione $c : V \to \{1,2,3\}$ tale che $c(u) \neq c(v)$ per ogni $(u,v) \in E$. Definiamo una 4-colorazione c' per $G' = f(G)$ nella maniera seguente: per ogni $v \in V' = V \cup \{x\}$

$$c'(v) = \begin{cases} c(v) & \text{se } v \in V \\ 4 & \text{altrimenti.} \end{cases}$$

Sia ora $(y,z) \in E'$ un arco di G': se $y \in V$ e $z \in V$ allora $c'(y) = c(y) \neq c(z) = c'(z)$ in quanto c è una colorazione per G; se invece $y \in V$ e $z = x$ allora $c(y) \in \{1,2,3\}$ e $c(z) = c(x) = 4$ (e analogamente per $y = x$ e $z \in V$). Dunque, c' è una colorazione per G'.
Supponiamo, infine, che G' sia 4-colorabile, ossia, che esista una funzione $c' : V \to \{1,2,3,4\}$ tale che $c(y) \neq c(z)$ per ogni $(y,z) \in E'$. Allora, poiché $(x,u) \in E'$ per ogni $u \in V$, $c'(x) \neq c(u)$ per ogni $u \in V$. Questo significa che gli elementi di V sono colorati mediante c' con 3 colori in modo tale che nodi adiacenti hanno colore differente. Dunque, G è 3-colorabile.
Questo completa la prova che f è una riduzione polinomiale da 3-COLORABILITÀ a 4-COLORABILITÀ.

Soluzione del problema 9.11

Il problema, che denoteremo, in breve, k-COL (mettendo in evidenza il ruolo di valore costante giocato da k), può essere formalizzato nella maniera seguente:

- $I_{k-COL} = \{G = (V,E) :\ G \text{ è un grafo non orientato }\}$.
- $S_{k-COL}(G) = \{V_1,\ldots,V_k : \forall i = 1,\ldots,k\ [\ V_i \subseteq V\ ]\ \}$.

- $\pi_{k-COL}(G, V_1, \ldots, V_k) = \forall i = 1, \ldots, k \; \forall u, v \in V_i \; [\; (u, v) \notin E \;] \; \wedge \; \cup_{i=1}^{k} V_i = V$.

Sia G un grafo istanza di 4-COL; la funzione f applicata a G trasforma il grafo G in un nuovo grafo G' costituito da due componenti connesse, G_1 e G_2: una di tali componenti, diciamo G_1 è il grafo G stesso, l'altra componente, diciamo G_2 è il grafo completo sui tre nodi a, b, c.

È evidente che il calcolo di $f(G)$ richiede tempo costante. Per verificare se f è una riduzione da 4-COL a 3-COL, occorre verificare se è vero che:

G è 4-colorabile se e soltanto se $f(G)$ è 3-colorabile.

Supponiamo che G sia 4-colorabile: allora, la componente G_1 di $f(G)$ è anch'essa 4-colorabile. Questo, però, non significa che $G_1 = G$ sia 3-colorabile: a titolo di esempio, consideriamo come grafo G il grafo completo di 4 nodi (che è 4-colorabile, ma non 3-colorabile). Quindi dalla 4-colorabilità di G non possiamo dedurre alcunché circa la 3-colorabilità di $f(G)$.

In conclusione, f non è una riduzione da 4-COL a 3-COL.

Soluzione del problema 9.12

Mostriamo di seguito la formalizzazione dei due problemi in questione, che indicheremo, in breve, rispettivamente, DS e EC:

- $I_{DS} = \{\langle G = (V, E), k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}\}$;
- $S_{DS}(G, k) = \{D \subseteq V\}$;
- $\pi_{DS}(G, k, D) = |D| \geq k \; \wedge \; \forall u \in V - D \; [\; \exists v \in D : (u, v) \in E \;]$

- $I_{EC} = \{\langle G = (V, E), k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}\}$;
- $S_{EC}(G, k) = \{E' \subseteq E\}$;
- $\pi_{EC}(G, k, E') = |E'| \geq k \; \wedge \; \forall (u, v) \in E - E' \; [\; \exists z \in V : (u, z) \in E' \; \vee \; (v, z) \in E' \;]$

Sia ora $\langle G = (V, E), k\rangle \in I_{EC}$ e sia $f(G, k) = \langle \overline{G} = (\overline{V}, \overline{E}), k\rangle$.
Se $\langle G = (V, E), k\rangle \in I_{EC}$ è una istanza sì per EC allora esiste $E' \subseteq E$, con $|E'| \leq k$, tale che, per ogni $(u, v) \in E - E'$, esiste $z \in V$ tale che $(u, z) \in E'$ o $(v, z) \in E'$. Definiamo, allora, $D = \{e \in \overline{V} = E : e \in E'\}$ e dimostriamo che esso è un Dominating set per $\overline{G}$. Sia $e \in \overline{V} - D$ con $e = (u, v)$; allora, per definizione di D, $e \in E - E'$ e quindi, poiché E' è un Edge Cover per G, esiste un arco $e' \in E'$ che è incidente su u oppure su v. Allora, $e' \in D$ e $(e, e') \in \overline{E}$. Questo prova che D è un Dominating Set per $\overline{G}$ e, poiché $|D| = |E'| \leq k$, questo prova che $f(G, k)$ è una istanza sì si DS.
Viceversa, se $f(G, k) = \langle \overline{G} = (\overline{V}, \overline{E}), k\rangle$ è una istanza sì di DS, allora esiste $D \subseteq \overline{V}$, con $|D| \leq k$, tale che, per ogni $e \in \overline{V} - D$ esiste $d \in D$ tale che $(e, d) \in \overline{E}$. Definiamo, allora, $E' = \{e \in E = \overline{V} : e \in D\}$ e dimostriamo che esso è unEdge Cover per G. Sia $e \in E - E'$; allora $e \in \overline{V} - D$ e quindi, poiché D è un Dominating Set per $\overline{V}$, esiste $d \in D$ tale che $(e, d) \in \overline{E}$. Ma $(e, d) \in \overline{E}$ significa che gli archi $e \in E$ e $d \in E$ hanno un estremo in comune, cioè, che l'arco e è coperto dall'arco d in G. Questo prova che E' è un Edge Cover per G e, poiché $|E'| = |D \leq k$, questo prova che $\langle G, k\rangle$ è una istanza sì si EC.
Poiché f è calcolabile in tempo polinomiale in $|G|$, questo prova che f è una riduzione polinomiale da EC a DS.

Per provare la **NP**-completezza di EC è necessario ridurre ad esso un problema noto **NP**-completo, e non ridurre esso ad un problema **NP**-completo (come fa la funzione f). Quindi, l'esistenza della funzione f non dimostra che EC è **NP**-completo.

Per completezza, osserviamo che, se interpretiamo f come una trasformazione di istanze di DS ad istanze di EC, essa non una riduzione da DS ad EC. Per provare questa affermazione, consideriamo un grafo $G = (V, E)$ in cui $V = \{u, v, z, x, y\}$ e $E = \{(u, v), (u, z), (u, x), (u, y)\}$ (stella centrata in u): è facile verificare che $D = \{u\}$ è un insieme dominante per G, ma che non esiste alcun Edge Cover di cardinalità 1 per $\overline{G}$. Infatti, $\overline{G}$ è una clique di 4 nodi (e, quindi, 6 archi) e per coprire tutti i suoi archi è necessario utilizzare un insieme di almeno 2 archi.

Soluzione del problema 9.13

Mostriamo di seguito la formalizzazione del problema indicato nel testo, che indicheremo, in breve, con l'acronimo PMN (PERCORSO SU METÀ NODI), e del problema HP:

- $I_{PMN} = \{\langle G = (V,E), u, v\rangle : G$ è un grafo non orientato e $u, v \in V\}$;
- $S_{PMN}(G,u,v) = \{\langle u, u_1, u_2, \ldots, u_h, v\rangle : \ \forall\, i = 1, \ldots, h\ [u_i \in V\ \wedge u_i \neq u\ \wedge u_i \neq v\ ]\ \}$;
- $\pi_{PMN}(G,k,\langle u, u_1, u_2, \ldots, u_h, v\rangle) = \left(h = \left\lceil \frac{|V|}{2} \right\rceil - 2\right) \wedge \forall i, j = 1, \ldots h : i \neq j\ [\ u_i \neq u_j\ ] \wedge\ \forall i = 1, \ldots h-1\ [\ (u_i, u_{i+1}) \in E\ ]\ \wedge\ (u, u_1) \in E\ \wedge (u_h, v) \in E$;

- $I_{HP} = \{\langle G = (V,E), u, v\rangle : G$ è un grafo non orientato e $u, v \in V\}$;
- $S_{HP}(G,u,v) = \{\langle u, u_1, u_2, \ldots, u_{|V|-2}, v\rangle : \ \forall\, i = 1, \ldots, h\ [u_i \in V\ \wedge u_i \neq u\ \wedge u_i \neq v\ ]\ \}$;
- $\pi_{HP}(G,k,\langle u, u_1, u_2, \ldots, u_{|V|-2}, v\rangle) = \forall i, j = 1, \ldots |V|-1 : i \neq j\ [\ u_i \neq u_j]\ \wedge\ \forall i = 1, \ldots |V|-3\ [\ (u_i, u_{i+1}) \in E\ ]\ \wedge\ (u, u_1) \in E\ \wedge (u_h, v) \in E$

Sia ora $\langle G = (V,E), u, v\rangle \in I_{HP}$ e sia $f(G,u,v) = \langle G' = (V \cup \overline{V}, E), u, v\rangle$, con $\overline{V} = \{\overline{u} : u \in V\}$, la trasformazione descritta nel testo. Banalmente, calcolare $f(G,u,v)$ richiede tempo in $\mathbf{O}(|V|)$.
Se $\langle G = (V,E), u, v\rangle \in I_{HP}$ è una istanza sì per HP allora esiste una sequenza $\langle u, u_1, u_2, \ldots, u_{|V|-2}, v\rangle$ di nodi distinti di V tale che $(u, u_1) \in E$, $(u_{|V|-2}, v) \in E$ e, per ogni $i = 1, \ldots, |V|-3$, $(u_i, u_{i+1}) \in E$. Per costruzione, $\langle u, u_1, u_2, \ldots, u_{|V|-2}, v\rangle$ è anche una sequenza di nodi distinti di V' e, inoltre, $(u, u_1) \in E'$, $(u_{|V|-2}, v) \in E'$ e, per ogni $i = 1, \ldots, |V|-3$, $(u_i, u_{i+1}) \in E'$. Infine,

$$|V| - 2 = \frac{2|V|}{2} - 2 = \frac{|V'|}{2} - 2.$$

Dunque, $langleG' = (V \cup \overline{V}, E), u, v\rangle$, è una istanza sì per PMN.
Viceversa, se $\langle G' = (V', E), u, v\rangle \in I_{HP}$ è una istanza sì per PMN allora esiste una sequenza $\langle u, u_1, u_2, \ldots, u_{|V|-2}, v\rangle$ di nodi distinti di V' tale che $(u, u_1) \in E$, $(u_{|V|-2}, v) \in E$ e, per ogni $i = 1, \ldots, |V|-3$, $(u_i, u_{i+1}) \in E$. Per costruzione, poiché i nodi in $\overline{V}$ sono isolati, $\langle u, u_1, u_2, \ldots, u_{|V|-2}, v\rangle$ è una sequenza di nodi distinti di V e, inoltre

$$\frac{|V'|}{2} - 2 = \frac{2|V|}{2} - 2 = |V| - 2.$$

Dunque, $langleG = (V,E), u, v\rangle$, è una istanza sì per HP.

Soluzione del problema 9.14

Il problema 2HDS può essere formalizzato come di seguito descritto:

- $I_{2HDS} = \{\langle G = (V,E), k\rangle : G$ è un grafo non orientato $\wedge\ k \in \mathbb{N}\}$;
- $S_{2HDS}(G,k) = \{H \subseteq V\ \}$;
- $\pi_{2HDS}(G,k,S_{2HDS}(G)) = \exists H \in S_{2HDS}(G,k) : \ |H| \leq k\ \wedge\ \forall\, u \in V - H\ \left[\ \exists v \in H,\ \exists z \in V : \ (u,z) \in E\ \wedge (z,v) \in E\ \right]$.

Cominciamo con l'osservare che tempo $\mathbf{O}(|E||V|$ è sufficiente per calcolare $f(G,k)$.

Sia, ora, $G = (V,E)$ il grafo costituito due nodi collegati da un arco: in questo caso, $V = \{u, v\}$ e $E = \{e = (u,v)\}$. Consideriamo l'istanza $\langle G, 1\rangle$ di DS; essa è una istanza sì, e gli unici insiemi dominanti costituiti da un singolo nodo sono gli insiemi $D_1 = \{u\}$ e $D_2 = \{v\}$.

L'istanza $f(G,1) = \langle G' = (V', E'), 1\rangle$ di 2HDS corrispondente a $\langle G, 1\rangle$ è, in questo caso, individuata dal grafo $G' = (V', E')$ tale che $V' = \{u, v, e\}$ e $E' = \{(u,e), (e,v)\}$ (si veda la Figura 9.3). In questo caso, mostriamo che nessun sottoinsieme H di V' costituito da un solo elemento soddisfa la condizione che deve essere soddisfatta da una soluzione effettiva di 2HDS.

Figura 9.3: Il grafo G, istanza di DOMINATING SET ed il suo corrispondente G', istanza di 2HDS, tramite la funzione f.

- Se $H = \{u\}$, il nodo e non soddisfa la condizione: infatti, l'unica coppia di nodi in $V' - \{e\}$ è u, v con $u \in H$ e $(e, v) \in E'$ ma $(v, u)\ \not{n}E'$.
- Se $H = \{v\}$, di nuovo il nodo e non soddisfa la condizione: infatti, l'unica coppia di nodi in $V' - \{e\}$ è u, v con $v \in H$ e $(e, u) \in E'$ ma $(u, v)\ \not{n}E'$.
- Se $H = \{e\}$, il nodo u e il nodo v non soddisfano entrambi la condizione: la verifica di questa affermazione è simile alle verifiche dei due casi precedenti ed è lasciata per esercizio.

Soluzione del problema 9.15

Il problema in esame, che indicheremo, in breve, con l'acronimo *SBC*, può essere formalizzato come di seguito descritto:

- $I_{SBC} = \{\langle G = (V, E), k\rangle : G$ è un grafo non orientato $\wedge\ k \in \mathbb{N}\}$;
- $S_{SBC}(G, k) = \{\langle\langle V_1, V_2\rangle :\ V_1 \subseteq V\ \wedge\ V_2 \subseteq V\ \wedge\ V_1 \cap V_2 = \emptyset\ \}\ \}$;
- $\pi_{SBC}(G, k, S_{3COL\vee SBC}(G, k)) = \exists \langle V_1, V_2\rangle \in S_{SBC}(G, k) :\ V_1 \cap V_2 = \emptyset\ \wedge\ |V_1 \cup V_2| \geq 2k\ \wedge\ \forall u \in V_1\ \forall v \in V_2\ [(u, v) \in E]$
$$\wedge\ \forall u, v \in V_1\ [(u, v) \notin E]\ \wedge\ \forall u, v \in V_2\ [(u, v) \notin E].$$

Ricordiamo che il problema CLIQUE (in breve, *CL*) può essere formalizzato come segue:

- $I_{CL} = \{\langle G = (V, E), k\rangle : G$ è un grafo non orientato $\wedge\ k \in \mathbb{N}\}$;
- $S_{CL}(G, k) = \{\langle V' \subseteq\ \}$;
- $\pi_{CL}(G, k, S_{CL}(G, k)) = \exists V' \in S_{SBC}(G, k) :\ |V'| \geq k\ \wedge\ \forall u, v \in V'\ [(u, v) \in E]\ .$

Dimostriamo che f è una riduzione polinomiale da CLIQUE a *SBC*. Sia $\langle G = (V, E), k\rangle$ una istanza *CL* e sia $f(G, k) = \langle \overline{G} = (\overline{V}, \overline{E}), k\rangle$. Osserviamo che, per costruzione, l'insieme $\overline{V}$ risulta naturalmente partizionato nei due sottoinsiemi $\overline{V}_1 = \{u_1 : u \in V\}$ e $\overline{V}_2 = \{u_2 : u \in V\}$.

→ Se $\langle G = (V, E), k\rangle$ è una istanza sì di *CL*, allora, esiste $V' \subseteq V$ tale che $|V'| \geq k$ e, per ogni $u, v \in V'$, $(u, v) \in E$. Allora, consideriamo i seguenti due sottoinsiemi U_1 e U_2 dell'insieme $\overline{V}$:
- $U_1 = \{u_1 \in \overline{V} : u \in V'\}$, e
- $U_2 = \{u_1 \in \overline{V} : u \in V'\}$.
Poiché $|V'| \geq k$, allora $|U_1 \cup U_2| \geq 2k$. Inoltre, per costruzione, per ogni coppia di nodi u_1 e v_1 in U_1, $(u_1, v_1) \notin \overline{E}$ e, per ogni coppia di nodi u_2 e v_2 in U_2, $(u_2, v_2) \notin \overline{E}$. Infine, poiché per ogni coppia di nodi u e v in V', $(u, v) \in E$, ancora per costruzione si ha che, per ogni coppia di nodi u_1 in U_1 e v_2 in U_2, $(u_1, v_1) \in \overline{E}$. Quindi, $U_1 \cup U_2$ induce

un sottografo bipartito completo di almeno $2k$ nodi in $\overline{G}$, e questo dimostra che $\langle\langle\overline{G} = (\overline{V},\overline{E}),k\langle$ è una istanza sì di *SBC*.

$\leftarrow$ Viceversa, se $f(G,k) = \langle\overline{G} = (\overline{V},\overline{E}),k\langle$ è una istanza sì di *SBC*, allora esistono due sottoinsiemi indipendenti U_1 e U_2 dell'insieme $\overline{V}$ tali che $|U_1 \cup U_2| \geq 2k$ e $U_1 \cup U_2$ induce un sottografo bipartito completo in $\overline{G}$. Senza perdita di generalità, possiamo assumere che $U_1 \subseteq \overline{V}_1$ e $U_2 \subseteq \overline{V}_2$: infatti, poiché $\overline{V}_1$ e $\overline{V}_2$ sono due insiemi indipendenti in $\overline{G}$ e poich'e U_1 e U_2 inducono un sottografo bipartito completo in $\overline{G}$, se esiste un nodo $x_1 \in U_2$ allora U_1 deve essere costituito da soli nodi in $\overline{V}_2$ (o x_1 non potrebbe essere adiacente a qualcuno di essi) e questo, a sua volta, per la stessa ragione, implica che U_2 deve essere costituito da soli nodi in $\overline{V}_1$; se questo accadesse, sarebbe allora sufficiente scambiare i due insiemi U_1 e U_2. Inoltre, ancora senza perdita di generalità, possiamo assumere che, per ogni $u \in V$,

$$u_1 \in U_1 \Leftrightarrow u_2 \in U_2;$$

infatti, se $u_1 \in U_1$, allora u_1 è adiacente a tutti i nodi in U_2 e quindi, per costruzione, u_2 è adiacente a tutti i nodi di U_1 e può, pertanto, essere inserito in U_2 lasciando vera la proprietà che $U_1 \cup U_2$ inducono in $\overline{G}$ un grafo bipartito completo e $|U_1 \cup U_2| \geq 2k$. Di conseguenza, $|U_1| = |U_2| \geq k$.
Sia V' l'insieme dei nodi in V che corrispondono a nodi in U_1 (o equivalentemente, per quanto appena osservato, in U_2), ossia,

$$V' = \{u \in V : u_1 \in U_1\} = \{u \in V : u_1 \in U_1\}.$$

Consideriamo due generici nodi u,v in V': poiché $u,v \in V'$, allora esistono $u_1,v_1 \in U_1$ e $u_2,v_2 \in U_2$; poiché $U_1 \cup U_2$ induce in $\overline{G}$ un sottografo bipartito completo, allora $(u_1,v_2) \in \overline{E}$ e $(u_2,v_1) \in \overline{E}$. Ma, per definizione della funzione f, se $(u_1,v_2) \in \overline{E}$ e $(u_2,v_1) \in \overline{E}$, allora $(u,v) \in E$. Siccome u e v sono due nodi generici in V', questo dimostra che, per ogni $u,v \in V'$, $(u,v) \in E$, ossia, V' induce in G un sottografo completo. Infine, poiché $|V'| = |U_1| = |U_2| \geq k$, allora $\langle G = (V,E),k\langle$ è una istanza sì di CLIQUE.

Quindi, f è una riduzione da CLIQUE a *SBC*. Infine, poiché costruire $\overline{G}$ a partire da G richiede tempo lineare in $|V|$ e $|E|$, questo dimostra che f è una riduzione polinomiale da CLIQUE a *SBC*.

Soluzione del problema 9.16

Il problema decisionale considerato, che chiameremo PARTIZIONE IN CLIQUE (in breve *PIC*), può essere formalizzato come di seguito descritto:

- $I_{PIC} = \{\langle G = (V,E),k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}\}$;
- $S_{PIC}(G,k) = \{\{V_1,\ldots,V_k\} : \forall\, i = 1,\ldots,k\ [\ V_i \subseteq V\ ]\ \wedge\ \cup_{i=1}^{k} V_i = V\ \wedge\ \forall\, i,j = 1,\ldots,k : i \neq j\ [\ V_i \cap V_j = \emptyset\ ]\}$;
- $\pi_{PIC}(G,k,S_{PIC}(G,k)) = \exists\{V_1,\ldots,V_k\} \in S_{PIC}(G,k) : \forall\, i = 1,\ldots,k\ \forall u,v \in V_i\ [\ (u,v) \in E\ ]$.

Ricordiamo, ora, che il problema COLORABILITÀ consiste nel chiedersi, dati un grafo non orientato $G = (V,E)$ ed un intero positivo k, se è possibile colorare ciascun nodo di G con uno di (al più) k colori possibili in modo tale che i nodi di ciascuna coppia di nodi adiacenti abbiano ricevuto colori diversi. Una delle possibili formalizzazioni del problema COLORABILITÀ è la seguente:

- $I_{COL} = \{\langle G = (V,E),k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}\}$;
- $S_{COL}(G,k) = \{\{V_1,\ldots,V_k\} : \forall\, i = 1,\ldots,k\ [\ V_i \subseteq V\ ]\ \wedge\ \cup_{i=1}^{k} V_i = V\ \wedge\ \forall\, i,j = 1,\ldots,k : i \neq j\ [\ V_i \cap V_j = \emptyset\ ]\}$;
- $\pi_{COL}(G,k,S_{COL}(G,k)) = \exists\{V_1,\ldots,V_k\} \in S_{COL}(G,k) : \forall\, i = 1,\ldots,k\ \forall u,v \in V_i\ [\ (u,v) \notin E\ ]$.

Osserviamo, ora, che istanze $\langle G = (V,E),k\rangle$ di *COL* in cui $k \geq |V|$ sono sempre, banalmente, istanze sì: infatti, in tal caso, è sufficiente colorare ciascun nodo del grafo con un colore diverso. Pertanto, il problema è **NP**-completo nel caso in cui $k < |V|$. Sia, dunque, $\langle G = (V,E),k\rangle$ una istanza di COLORABILITÀ tale che $k < |V|$ e sia $f(G,k) = \langle\overline{G} = (V,\overline{E}),k\rangle$ l'istanza corrispondente di *PIC*.

Se $\langle G=(V,E),k\rangle$ è una istanza sì di COLORABILITÀ, allora esiste una partizione di V in k insiemi $V_1,\ldots,V_k$ tale che, per ogni $i=1,\ldots,k$ e per ogni coppia di nodi u e v contenuti entrambi in V_i, si ha che $(u,v)\notin E$. Ma, per definizione di $\overline{E}$, questo significa che, per ogni $i=1,\ldots,k$ e per ogni coppia di nodi u e v contenuti entrambi in V_i, si ha che $(u,v)\in\overline{E}$: quindi per ogni $i=1,\ldots,k$, V_i è un sottografo completo di $\overline{G}$. In conclusione, $\{V_1,\ldots,V_k\}$ è una partizione di $\overline{G}$ in k sottografi completi e questo significa che $\langle\overline{G}=(V,\overline{E}),k\rangle$ è una istanza sì di *PIC*.
Viceversa, se $\langle\overline{G}=(V,\overline{E}),k\rangle$ è una istanza sì di *PIC*, allora esiste una partizione di V in k insiemi $V_1,\ldots,V_k$ tale che, per ogni $i=1,\ldots,k$ e per ogni coppia di nodi u e v contenuti entrambi in V_i, si ha che $(u,v)\in\overline{E}$. Ma, per definizione di $\overline{E}$, questo significa che, per ogni $i=1,\ldots,k$ e per ogni coppia di nodi u e v contenuti entrambi in V_i, si ha che $(u,v)\notin E$: quindi per ogni $i=1,\ldots,k$, V_i è un insieme indipendente in G ed i suoi nodi possono essere colorati con lo stesso colore. Quindi, G può essere colorato con k colori e, dunque, $\langle G=(V,E),k\rangle$ è una istanza sì di COLORABILITÀ.
Questo dimostra che f è una riduzione da COLORABILITÀ a *PIC*.

Per calcolare f è sufficiente calcolare l'insieme $\overline{E}$ e, quindi, considerare tutte le coppie di nodi e, per ciascuna di esse, inserire l'arco corrispondente in $\overline{E}$ se e soltanto se esso non è in E. L'algoritmo che calcola f è pertanto descritto nel seguente frammento di codice:

```
Ē ← ∅;
for all (u ∈ V) do begin
    for all (v ∈ V) do begin
        if (u ≠ v ∧ (u,v) ∉ E) then Ē ← Ē ∪ {(u,v)};
    end;
end.
```

L'algoritmo appena descritto richiede $\mathbf{O}(|V|^2|E|)$ passi e, quindi, calcolare f richiede tempo polinomiale in $|G|$. Questo termina la prova che f è una riduzione polinomiale da COLORABILITÀ a *PIC*.

Soluzione del problema 9.17

Il problema decisionale considerato, che chiameremo A, può essere formalizzato come di seguito descritto:

- $I_A=\{\langle G=(V,E),k\rangle : G$ è un grafo non orientato e $k\in\mathbb{N}\}$;
- $S_A(G,k)=\{V' :\subseteq V\ \}$;
- $\pi_A(G,k,S_A(G,k))=\exists V'\in S_A(G,k) : \forall u\in V-V'\ \forall(u,v)\in E\ [\ v\in V'\ ]$.

Osserviamo che il predicato π_A del problema può essere espresso nella maniera seguente:

$$\exists V'\in S_A(G,k) : \forall(u,v)\in E\ [\ u\in V' \vee v\in V'\ ],$$

e, quindi, il problema A coincide con il problema VERTEX COVER.

L'algoritmo richiesto nel testo, che riceve in input un intero k, la matrice di adiacenza M di un grafo $G=(V,E)$ e l'insieme P di tutti i sottoinsiemi di V, è descritto nel seguente frammento di codice:

```
trovato ← falso;
while (P ≠ ∅∧ trovato= falso) do begin
    estrai un elemento V′ da P:
    if (|V′| ≤ k) then begin
        trovato ← vero;
        for (u ∈ V − V′) do
            for (v ∈ V) do
                if (M[u,v] = 1 wedge v ∉ V′) then trovato ← falso;
    end;
end.
```

Analizziamo, ora, la complessità del frammento di codice appena descritto.
Poiché accedere ad un elemento della matrice M ha costo costante, e verificare se un nodo è in V' ha costo proporzionale a $|V'|$, l'istruzione **if** alla linea 8 ha costo $\mathbf{O}(|V|)$; pertanto, il doppio ciclo **for** alle linee 6 e 7 ha costo $\mathbf{O}(|V|^3)$.
Testare la condizione dell'istruzione **if** alla linea 4 ha costo $\mathbf{O}(k|V|) \subseteq \mathbf{O}(|V|^2)$ (si osservi, infatti, che $k \leq |V|$).
Il numero di iterazioni del ciclo **while** alla linea 2 è $|P|$, e, quindi, il costo del frammento di codice è $\mathbf{O}(|P| \cdot |V|^2)$, ossia, è polinomiale *nella dimensione dell'input* o, in altri termini, è polinomiale in $|\chi(G)|$.

Osserviamo, infine, che la codifica $\chi(G)$ non è una codifica ragionevole di G: infatti, $|P| = 2^{|V|}$ e, quindi, la codifica di G mediante la sola matrice di adiacenza (che codifica tutte le informazioni necessarie ad individuare un grafo) è esponenzialmente più corta di $\chi(G)$. Poiché un problema è in **P** se esiste un algoritmo deterministico che richiede tempo polinomiale nella dimensione di una *codifica ragionevole* delle sue istanze, il frammento di codice non permette di affermare che il problema A è in **P** (e, in effetti, esso coincide con VERTEX COVER ed è, quindi, **NP**-completo).

Soluzione del problema 9.18

Il problema decisionale considerato, che chiameremo PARTIZIONE IN 3 INSIEMI INDIPENDENTI (in breve, $P3II$), può essere formalizzato come di seguito descritto:

- $I_{P3II} = \{G = (V,E) : G \text{ è un grafo non orientato } \}$;
- $S_{P3II}(G) = \{\langle V_1, V_2, V_3 \rangle : \subseteq V_1, V_2, V_3 \subseteq V \ \}$;
- $\pi_{P3II}(G, S_{P3II}(G)) = \exists \langle V_1, V_2, V_3 \rangle \in S_{P3II}(G) : \ V_1 \cup V_2 \cup V_3 = V \ \wedge \ V_1 \cap V_2 = \emptyset \ \wedge \ V_1 \cap V_3 = \emptyset \ \wedge$
$V_2 \cap V_3 = \emptyset \ \wedge \ \forall i = 1,2,3 \ \forall u, v \in V_i \ [\ (u,v) \not\in E \ ]$.

Osserviamo che l'insieme P nella codifica di un grafo G è un sottoinsieme di $S_{PII}(G)$. In particolare, il predicato π_{P3II} del problema può essere espresso nella maniera seguente:

$$\exists \langle V_1, V_2, V_3 \rangle \in P : \forall i = 1,2,3 \ \forall u, v \in V_i \ [\ (u,v) \not\in E \ ],$$

Quindi, l'algoritmo richiesto nel testo, che riceve in input la matrice di adiacenza M di un grafo $G = (V,E)$ e l'insieme P di tutte le partizioni di V in insiemi indipendenti in G, è descritto nel seguente frammento di codice, che restituisce `vero` se G è una istanza sì di $P3II$:

```
1   trovato ← falso;
2   while (P ≠ ∅ ∧ trovato= falso) do begin
3       estrai un elemento ⟨V1, V2, V3⟩ da P:
4       trovato ← vero;
5       for (i ← 1; i ≤ 3; i ← i+1) do
6           for (u ∈ Vi) do
7               for (v ∈ Vi) do
8                   if (M[u,v] = 1) then trovato ← falso;
9   end;
10  Output: trovato.
```

Analizziamo, ora, la complessità del frammento di codice appena descritto.
Poiché accedere ad un elemento della matrice M ha costo costante, l'istruzione **if** alla linea 8 ha costo costante; pertanto, poiché, per ogni $\langle V_1, V_2, V_3 \rangle \in P$, la cardinalità di V_1, $|V_2|$ e $|V_3|$ è al più $|V|$, il doppio ciclo **for** alle linee 6 e 7 ha costo $\mathbf{O}(|V|^2)$.
Il numero di iterazioni del ciclo **for** alla linea 5 è costante, il numero di iterazioni del ciclo **while** alla linea 2 è $|P|$, e, quindi, il costo del frammento di codice è $\mathbf{O}(|P| \cdot |V|^2)$, ossia, è polinomiale *nella dimensione dell'input* o, in altri termini, è polinomiale in $|\chi(G)|$.

Osserviamo, infine, che la codifica $\chi(G)$ non è una codifica ragionevole di G: infatti, $|P| = 3^{|V|}$ e, quindi, la codifica di G mediante la sola matrice di adiacenza (che codifica tutte le informazioni necessarie ad individuare un grafo e che ha dimensione $|V|^2)$ è esponenzialmente più corta di $\chi(G)$. Poiché un problema è in **P** se esiste un algoritmo deterministico che richiede tempo polinomiale nella dimensione di una *codifica ragionevole* delle sue istanze, il frammento di codice non permette di affermare che il problema A è in **P** (e, in effetti, esso coincide con COLORABILITÀ ed è, quindi, **NP**-completo).

Soluzione del problema 9.19

Il frammento di codice di cui al punto 2.a) non è un algoritmo non deterministico in quanto l'operazione non deterministica **scegli** può effettuare una scelta all'interno di un insieme di dimensione *costante*, mentre nel frammento di codice del punto 2.a) l'insieme in cui viene effettuata la scelta è l'insieme $\{2,\ldots,n-1\}$ la cui dimensione è funzione dell'istanza e, pertanto, non costante.

Nel caso peggiore (che si presenta quando n è un numero primo), il frammento di codice di cui al punto 2.b) esegue $(n-2)^2$ volte l'istruzione alla linea 4. Dunque, esso richiede tempo $\geq (n-2)^2$. Poiché la codifica binaria del numero n richiede spazio $\lceil \log_2 n \rceil$, allora per qualunque codifica ragionevole di n si ha che $|n| = \mathbf{O}(\log_2 n)$. Conseguentemente, il tempo richiesto dal frammento di codice è

$$\geq (n-2)^2 = \mathbf{O}(2^{|n|}).$$

Il frammento di codice, dunque, opera in tempo pseudopolinomiale, ma non in tempo polinomale.

La discussione sopra dei due punti 2.a) e 2.b) non ci permette di trarre alcuna conclusione circa l'appartenenza del problema alla classe **P** o alla classe **NP**.
Tuttavia, possiamo osservare che un certificato per una istanza del problema è una coppia di interi $\langle h,k\rangle$: poiché $|h| \leq |n|$ e $|k| \leq |n|$, e poiché per verificare se un certificato è una soluzione effettiva è sufficiente calcolare il prodotto hk e tale operazione è eseguibile in tempo polinomiale in $|h|$ e $|k|$, possiamo concludere che il problema appartiene alla classe **NP**.

2.2 Appartenenza a NPC

Soluzione del problema 9.20

Il problema PARTIZIONE IN INSIEMI INDIPENDENTI (in breve, ISP) è formalizzato nella maniera seguente:

- $\mathrm{I}_{ISP} = \{\langle G=(V,E),k\rangle : G$ è un grafo non orientato e k è un intero positivo $\}$.
- $\mathrm{S}_{ISP}(\langle G=(V,E),k\rangle) = \{\langle V_1,\ldots,V_h\rangle : \forall 1 \leq i,j \leq h[V_i \cap V_j = \emptyset \wedge \cup_{i=1}^{h} V_i = V]\}$.
- $\pi_{ISP}(\langle G=(V,E),k\rangle,\langle V_1,\ldots,V_h\rangle) = \forall 1 \leq i \leq h[u,v \in V_i \Rightarrow (u,v) \notin E] \ \wedge \ h \leq k$.

Osserviamo ora che la definizione di tale problema coincide con quella del problema COLORABILITÀ: dunque, la NP-completezza di ISP segue immediatamente dalla NP-completezza di COLORABILITÀ, essendo i due problemi coincidenti (volendo, è possibile definire una riduzione polinomiale da COLORABILITÀ ad ISP la cui funzione di trasformazione da istanze di COLORABILITÀ ad istanze di ISP è la funzione identità).

Soluzione del problema 9.21

Il problema PARTIZIONE IN CLIQUE (in breve, PIC) è formalizzato nella maniera seguente:

- $\mathrm{I}_{PIC} = \{\langle G=(V,E),k\rangle : G$ è un grafo non orientato e k è un intero positivo $\}$.
- $\mathrm{S}_{PIC}(\langle G=(V,E),k\rangle) = \{\langle V_1,\ldots,V_h\rangle : \forall 1 \leq i,j \leq h[V_i \cap V_j = \emptyset \wedge \cup_{i=1}^{h} V_i = V] \wedge h \leq k\}$.
- $\pi_{ISP}(\langle G=(V,E),k\rangle,\langle V_1,\ldots,V_h\rangle) = \forall 1 \leq i \leq h[u,v \in V_i \Rightarrow (u,v) \in E] \ \wedge \ h \leq k$.

Il problema è in NP: infatti, una partizione dell'insieme dei nodi in k sottoinsiemi può essere generata non deterministicamente in tempo O(n). Indicato con $V = \{v_1, v_2, \ldots, v_n\}$ l'insieme dei nodi di G, si consideri allo scopo un albero delle computazioni di profondità n e grado di non determinismo k in cui il livello $j = 0, 2, \ldots, n-1$ dell'albero corrisponde al nodo v_{j+1} di G e l'i-esimo arco uscente da un nodo del livello j corrisponde all'inserimento di v_{j+1} in V_i.

Per quanto riguarda la completezza, essa viene dimostrata mediante una semplice riduzione dal problema COLORABILITÀ. Si consideri un'istanza $\langle G = (V,E), k\rangle$ di COLORABILITÀ e si costruisca il grafo complemento di G $G^c = (V, E^c)$ tale che, per ogni coppia di nodi $u, v \in V$, $(u,v) \in E^c$ se e soltanto se $(u,v) \notin E$. Il grafo G^c è facilmente calcolabile in tempo in O($|G|$).

Consideriamo una istanza $\langle G = (V,E), k\rangle$ di COLORABILITÀ: allora, $\langle G = (V,E), k\rangle \in$ COLORABILITÀ **se e soltanto se** esiste una partizione di V in $h \leq k$ sottoinsiemi $V_1, V_2, \ldots, V_h$ tali che, per ogni $i = 1, 2, \ldots, h$ e per ogni coppia di nodi $u, v \in V_i$, allora $(u,v) \notin E$. Ma questo significa che, per ogni $i = 1, 2, \ldots, h$ e per ogni coppia di nodi $u, v \in V_i$, allora $(u,v) \in E^c$. E questo è vero **se e soltanto se** $\langle G^c = (V, E^c), k\rangle \in$ PIC.

La dimostrazione di NP-completezza è così completata.

Soluzione del problema 9.22

Formalizzazione del problema:

- $I = \{\langle X = \{x-1, \ldots, x_n\}, f(X) = c_1 \wedge c_2 \wedge \ldots \wedge c_m\rangle : X$ è un insieme di variabili booleane e, per ogni $1 \leq j \leq m$, $c_j = l_{j1} \vee l_{j2} \vee l_{j3} \vee l_{j4}$ con $l_{ji} \in X$ oppure $\neg l_{ij} \in X$ per $i = 1, 2, 3, 4\}$;
- $S(X, f) = \{a : X \rightarrow \{$ **vero, falso** $\}\}$;
- $\pi(X, f, a) = f(a(X))$.

Una istanza $\langle X, f\rangle$ del problema 4-SODDISFACIBILITÀ è anche istanza del problema SODDISFACIBILITÀ: segue banalmente da questa osservazione che il problema 4-SODDISFACIBILITÀ è contenuto nella classe NP (formalmente, 4-SODDISFACIBILITÀ$\leq$SODDISFACIBILITÀ).

Per dimostrarne la completezza, mostriamo una riduzione polinomiale dal problema 3-SODDISFACIBILITÀ. Ricordiamo che il problema 3-SODDISFACIBILITÀ consiste nel chiedersi se una funzione booleana in forma congiuntiva normale con clausole di esattamente 3 letterali ciascuna è soddisfacibile.

Sia $X = \{x_1, \ldots, x_n\}$ un insieme di variabili booleane e $g(X) = d_1 \wedge \ldots \wedge D_m\rangle : X$ dove, per ogni $1 \leq j \leq m$, $d_j = l_{j1} \vee l_{j2} \vee l_{j3}$ con $l_{ji} \in X$ oppure $\neg l_{ji} \in X\}$ (ossia, $\langle X, g\rangle$ è un'istanza di 3-SODDISFACIBILITÀ). Costruiamo, a partire da $\langle X, g\rangle$, un'istanza $\langle X \cup Y, f(X \cup Y, g) = c_1 \wedge c'_1 \wedge c_2 \wedge c'_2 \wedge \ldots \wedge c_m \wedge c'_m\rangle$ di 4-SODDISFACIBILITÀ nella maniera seguente:

- definiamo l'insieme di variabili booleane $Y = \{y_1, \ldots, y_m\}$, contenente una variabile booleana per ciascuna clausola in g;
- per $j = 1, \ldots, m$, associamo alla clausola $d_j = l_{j1} \vee l_{j2} \vee l_{j3}$ (con $l_{ji} \in X$ oppure $\neg l_{ij} \in X$) di g la coppia di clausole $c_j = l_{j1} \vee l_{j2} \vee l_{j3} \vee y_j$ e $c'_j = l_{j1} \vee l_{j2} \vee l_{j3} \vee \neg y_j$.

Per costruzione, $\langle X \cup Y, f = c_1 \wedge c'_1 \wedge \ldots \wedge c_m \wedge c'_m\rangle$ è un'istanza di 4-SODDISFACIBILITÀ. Inoltre, essa può essere calcolata in tempo in $O(m)$ a partire da g. Resta da mostrare che che g è soddisfacibile se e soltanto se f è soddisfacibile:

1. se g è soddisfacibile allora esiste una assegnazione di verità a per le variabili in X che soddisfa ciascuna clausola $d_j$in g: poiché $c_j = d_j \vee y_j$ e $c'_j = d_j \vee \neg y_j$ allora a soddisfa sia c_j che c'_j qualunque sia l'assegnazione di verità alle variabili in Y;

2. se f è soddisfacibile allora esiste una assegnazione di verità b per le variabili in $X \cup Y$ che soddisfa ciascuna clausola c_j e ciascuna clausola c'_j in f. Mostriamo ora che, per ogni $j = 1, \ldots, m$, $b(X)$ soddisfa ciascuna clausola d_j: infatti, se $b(y_j) =$**vero** allora

$$\textbf{vero} = b(c'_j) = b(d_j \vee \neg y_j) = b(d_j) \vee b(\neg y_j) = b(d_j) \vee \textbf{ falso } = b(d_j),$$

e se $b(y_j) =$**falso** allora

$$\textbf{vero} = b(c_j) = b(d_j \vee y_j) = b(d_j) \vee b(y_j) = b(d_j) \vee \textbf{ falso } = b(d_j).$$

Allora, b è una assegnazione di verità che soddisfa g.

Si osservi che questa trasformazione (e la prova della sua correttezza) è la stessa che trasforma clausole di 2 letterali in clausole di 3 letterali nella riduzione da SODDISFACIBILITÀ a 3-SODDISFACIBILITÀ.

Soluzione del problema 9.23

Il problema LONGEST PATH (in breve, LP) è formalizzato nella maniera seguente:

- $I_{LP} = \{\langle G = (V,E), u, v, k \rangle : G$ è un grafo non orientato, $u \in V$, $v \in V$, e k è un intero positivo $\}$.
- $S_{LP}(\langle G = (V,E), u, v, k \rangle) = \{p = \langle v_1, v_1, \ldots, v_h \rangle : \forall 1 \leq i \leq h [v_i \in V]\}$.
- $\pi_{LP}(\langle G = (V,E), u, v, k \rangle, p = \langle v_1, \ldots, v_h \rangle) = \forall 0 \leq i < h[(v_i, v_{i+1} \in E)] \wedge u = v_1 \wedge v = v_h \wedge h \geq k+1$.

Il problema è in **NP**: infatti, un qualunque sottoinsieme dell'insieme dei nodi può essere generato non deterministicamente in tempo O(n). Un tale sottoinsieme, se di cardinalità h, corrisponde a $h!$ percorsi possibili in G, ove ciascun percorso corrisponde ad una permutazione degli elementi del sottoinsieme: ad esempio, se consideriamo il sottoinsieme $\{u, v, w, z\}$ di V, i percorsi possibili ad esso corrispondenti sono $\langle u, v, w, z \rangle$, $\langle u, v, z, w \rangle$, $\langle u, w, v, z \rangle$, $\langle u, w, z, v \rangle$,$\langle u, z, v, w \rangle$, $\langle u, z, w, v \rangle$, e così via. Pertanto, possiamo descrivere l'algoritmo non deterministico che decide LP nella maniera seguente:

$V' \leftarrow \emptyset$;
Fase 1: generazione di V'
    for ($x \in V$) **do**
        scegli se inserire o meno x in V';

Fase 2: generazione di una permutazione degli elementi di V'
    $i \leftarrow 1$;
    while ($V' \neq \emptyset$) **do begin**
        scegli $x \in V'$;
        $v_i \leftarrow x$;
        $V' \leftarrow V' - \{x\}$;
        $i \leftarrow i+1$;
    end
    ora $\langle v_1, v_2, \ldots v_{|V'|} \rangle \in S_{LP}(G, u, v, k)$

Fase 3: verifica del predicato π_{LP}
    $\pi \leftarrow vero$;
    if ($v_1 \neq u \vee v_{|V'|} \neq v \vee |V'| < k+1$) **then** $\pi \leftarrow falso$;
    $i \leftarrow 1$;
    while ($i < |V'| \wedge \pi = vero$) **do begin**
        if ((v_i, v_{i+1}) $\notin E$) **then** $\pi \leftarrow falso$;
        $i \leftarrow i+1$;

end

if ($\pi = vero$) **then Output**: accetta;
else Output: rigetta;

Poiché la Fase 1 e la Fase 2 dell'algoritmo richiedono tempo in O(n) e la Fase 3 richiede tempo in O($n|E|$), tale algoritmo opera in tempo (non deterministico) polinomiale.

Dimostriamo, ora, la completezza di LP rispetto ad **NP** mediante una riduzione polinomiale dal problema PERCORSO HAMILTONIANO (in breve, HP).
Sia $\alpha = \langle G = (V,E), u, v \rangle$ un'istanza di HP: trasformiamo tale istanza nell'istanza $\beta = \langle G = (V,E), u, v, |V|-1 \rangle$ di LP. Tale traformazione richiede tempo polinomiale in $|\alpha|$.
Supponiamo che $\alpha \in$HP: allora, G contiene un cammino da u a v che passa per ogni nodo in V una e una sola volta e, dunque, ha lunghezza $|V|-1$. Questo significa che $\beta \in$LP.
Viceversa, supponiamo che $\beta \in$HP: allora, G contiene un cammino da u a v che ha lunghezza $|V|-1$ e, dunque, passa necessariamente per ogni nodo in V una e una sola volta. Questo significa che $\alpha \in$LP.

Mostriamo, infine, una riduzione polinomiale dal problema CIRCUITO HAMILTONIANO (in breve, HC) al problema HP.
Sia $G = (V,E)$ un'istanza di HC, con $V = \{v_1, v_2, \ldots, v_n\}$; trasformiamo tale istanza nell'istanza $\alpha = \langle G_P = (V_P, E_P), a, b \rangle$ come di seguito specificato:

- a e b sono due nuovi nodi, non contenuti in V;
- $V_P = V \cup \{a,b\}$;
- $E_P = E \cup E_{ab}$, dove

$$E_{ab} = \{(a, v_1)\} \cup \{(b, v_i) : v_i \in V \wedge (v_1, v_i) \in E\}.$$

Supponiamo che $G \in$HC: allora, G contiene un ciclo $\langle v_1, v_{i_2}, \ldots, v_{i_n} \rangle$ che tocca tutti i nodi una ed una sola volta. Allora, $(v_1, v_{i_n}) \in E$ e, quindi, $(b, v_{i_n}) \in E_P$. Segue da ciò che $\langle a, v_1, v_{i_2}, \ldots, v_{i_n}, b \rangle$ è un percorso hamiltoniano in G_P da a a b e, quindi, $\alpha \in$HP.
Viceversa, supponiamo che $\alpha \in$HP: allora, G_P contiene un percorso hamiltoniano da a a b. Poiché l'unico nodo adiacente ad a in G_P è v_1, tale percorso sarà del tipo $\langle a, v_1, v_{i_2}, \ldots, v_{i_n}, b \rangle$; poiché $(v_{i_n}, b) \in E_P$, segue dalla definizione di E_{ab} che $(v_1, v_{i_n}) \in E$. Quindi, $\langle v_1, v_{i_2}, \ldots, v_{i_n} \rangle$ è un ciclo hamiltoniano in G.

Soluzione del problema 9.24

Segue dalla definizione di colorabilità di un grafo che

due nodi possono essere colorati con lo stesso colore solo se essi non sono adiacenti.

Allora, il problema in questione (che sarà denotato, in breve, SUB-1COL) può essere formalizzato nella maniera seguente:

- $I_{SUB-1COL}$ = $\{\langle G = (V,E), k \rangle :$ G è un grafo non orientato e k è un intero positivo $\}$.
- $S_{SUB-1COL}(G,k) = \{V' \subseteq V\}$.
- $\pi_{SUB-1COL}(G,k,V') = |V'| = k \wedge [\forall u, v \in V' : (u,v) \notin E]$.

Si osservi che la definizione di tale problema coincide con quella del problema INSIEME INDIPENDENTE. Dunque, il problema SUB-1COL=INDEPENDENT SET è **NP**-completo.

Soluzione del problema 9.25

Osserviamo, innanzi tutto, che il problema è in **NP**: infatti, possiamo generare in tempo non deterministico polinomiale tutte le assegnazioni di verità per l'insieme X (allo stesso modo in operiamo per il problema SAT) e poi, per ciascuna di esse, possiamo verificare il predicato π_{+2SAT} in tempo deterministico polinomiale.

Per quanto concerne la completezza, mostriamo ora che +2SAT $\leq$ VERTEX COVER (VC). Si ricordi che un'istanza di VC è una coppia $\langle G = (V,E), k \rangle$, in cui G è un grafo non orientato e k un intero positivo, che una soluzione possibile è un sottoinsieme $V' \subseteq V$ di al più k nodi di G, e che una soluzione possibile V' è una soluzione effettiva se, per ogni arco $(u,v) \in E$, $u \in V'$ o $v \in V'$.
Sia data, dunque, un'istanza $\langle G = (V,E), k \rangle$ di VC. Associamo a G la formula booleana f_G nel modo seguente:

- $X = \{x_u : u \in V\}$;
- per ogni arco $e = (u,v) \in E$, definiamo la clausola $c_e = x_u \vee x_v$;
- f_G è la congiunzione di tutte le clausole c_e: se $E = \{e_1, e_2, \ldots e_m\}$, allora

$$f_G(X) = \bigwedge_{e \in E} c_e = c_{e_1} \wedge c_{e_2} \wedge \ldots \wedge c_{e_m}.$$

Osserviamo che f_G rispetta i vincoli definiti per +2SAT. Dunque, $\langle f_G, k \rangle \in I_{+2SAT}$.
Supponiamo che $G = (V,E)$ ammetta un Vertex Cover di al più k nodi: allora, esiste $V' \subseteq V$ con $|V'| \leq k$ tale che, per ogni $(u,v) \in E$, $u \in V' \vee v \in V'$. Definiamo, allora, la seguente assegnazione di verità a per X: per ogni $u \in V'$ $a(x_u) = vero$, $u \notin V'$ $a(x_u) = falso$. Poiché $|V'| \leq k$, segue che $|\{x \in X : a(x) = vero\}| \leq k$; inoltre, poiché una clausola in f_G corrisponde ad un arco in G e ogni arco ha almeno un estremo in V', allora ogni clausola in f_G è soddisfatta da a, ossia, $f_G(a(X)) = vero$.
Supponiamo, ora, che esista una assegnazione di verità a per X che soddisfa f_G e tale che $|\{x \in X : a(x) = vero\}| \leq k$. Definiamo, allora, $V' = \{u \in V : a(x_u) = vero\}$. Per costruzione, $|V'| \leq k$; inoltre, poiché ad ogni arco in E corrisponde una clausola in f_G ed ogni clausola contiene almeno un letterale vero, allora ogni arco in E ha almeno un estremo in V', ossia, V' è un Vertex Cover per G.

Abbiamo quindi dimostrato che $\langle G = (V,E), k \rangle \in$ VC se e solo se $\langle f_G, k \rangle \in$ +2SAT. Poiché la costruzione di f_G richiede tempo polinomiale, il problema +2SAT è **NP**-completo.

Soluzione del problema 9.26

Il problema, che denoteremo, in breve, BVC (BOUNDED SIZE VERTEX COVER), può essere formalizzato nella maniera seguente:

- $I_{BVC} = \{\langle G = (V,E), k \rangle :\ G$ è un grafo non orientato $\}$.
- $S_{BVC}(G,k) = \{V' \subseteq V\}$.
- $\pi_{BVC}(G,k,V') = |V| = k \ \vee \ \{ \ [\forall (u,v) \in E(\ u \in V' \vee v \in V' \ )] \ \wedge \ |V'| \leq k \ \}$.

Il problema BVC coincide con il problema VERTEX COVER. Infatti, consideriamo un grafo $G = (V,E)$ ed un intero k:

- se G è un grafo di k nodi, allora V è un Vertex Cover per G e, quindi, $\langle G = (V,E), k \rangle$ è una istanza sì sia per BVC che per VC;
- se $|V| > k$, allora $\langle G = (V,E), k \rangle$ è una istanza sì per BVC se e soltanto se $\langle G = (V,E), k \rangle$ è una istanza sì per VC.

Quindi, il problema è **NP**-completo.

Soluzione del problema 9.27

Il problema, che denoteremo, in breve, SC (SPECIAL COLORABILITY), può essere formalizzato nella maniera seguente:

- $I_{SC} = \{\langle G=(V,E),k\rangle :\ G$ è un grafo non orientato $\}$.
- $S_{SC}(G,k) = \{c : V \to \{1,\ldots,k\}\ \}$.
- $\pi_{SC}(G,k,c) = \forall u,v \in V\ [\ c(u) < k \wedge c(v) < k \wedge c(u) \neq c(v) \Rightarrow\ (u,v) \notin E\ ]$.

Riduciamo problema INDEPENDENT SET (IS) al problema SC. Allo scopo, consideriamo una istanza $\langle G=(V,E),k\rangle$ di IS: l'istanza corrispondente di SC è $\langle G'=(V',E'),k+1\rangle$ in cui, semplicemente, viene incrementato il valore numerico e viene aggiunto un nodo all'insieme V collegandolo con tutti gli elementi di V: ossia, $V' = V \cup \{x\}$ (con $x \notin V$) e $E' = E \cup \{(x,u) : u \in V\}$. Mostriamo ora che $\langle G=(V,E),k\rangle$ è una istanza sì per IS se e soltanto se $\langle G'=(V',E),k+1\rangle \in$ è una istanza sì per SC.

- Se G contiene un insieme indipendente I di k nodi, allora coloriamo in G' ciascun elemento di I con uno dei primi k colori (nodi distinti avranno colori distinti) ed i nodi in $V'-I$ con il colore k. Poiché $x \notin V$, allora $x \in V'-I$, ossia, V' è non vuoto, e, dunque, tutti i $k+1$ colori sono effettivamente utilizzati. Tale colorazione rispetta i vincoli di SC e, quindi, $\langle G=(V,E),k\rangle$ è una istanza sì per SC.
- Se G non contiene un insieme indipendente di k nodi, allora non è possibile trovare in G k nodi distinti cui assegnare i primi k colori; inoltre, poiché x è collegato a tutti i nodi in V, x non può essere colorato con uno dei primi k colori. Quindi, $\langle G=(V,E),k\rangle$ è una istanza no per SC.

Quindi, il problema è **NP**-completo.

Soluzione del problema 9.28

Mostriamo di seguito la formalizzazione del problema in questione, che indicheremo, in breve, $\{1,2\}-1$-COL:

- $I_{\{1,2\}-1-COL} = \{\langle G=(V,E),k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}\}$;
- $S_{\{1,2\}-1-COL}(G,k) = \{V' \subseteq V\}$;
- $\pi_{\{1,2\}-1-COL}(G,k,V') = |V'| \geq k\ \wedge\ \forall (u,v) \in E\ [\ u \notin V' \vee v \notin V'\ ]$,

in cui V' è l'insieme dei nodi colorati con il colore 1.
A questo punto, è sufficiente osservare che il problema $\{1,2\}-1$-COL coincide con il problema INDEPENDENT SET per provarne la **NP**-completezza.

Problema 3. Mostriamo di seguito la formalizzazione del problema in questione, che indicheremo, in breve, $\{1,2\}-1$-COL:

- $I_{\{1,2\}-1-COL} = \{\langle G=(V,E),k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}\}$;
- $S_{\{1,2\}-1-COL}(G,k) = \{V' \subseteq V\}$;
- $\pi_{\{1,2\}-1-COL}(G,k,V') = |V'| \geq k\ \wedge\ \forall (u,v) \in E\ [\ u \notin V' \vee v \notin V'\ ]$,

in cui V' è l'insieme dei nodi colorati con il colore 1.
A questo punto, è sufficiente osservare che il problema $\{1,2\}-1$-COL coincide con il problema INDEPENDENT SET per provarne la **NP**-completezza.

Soluzione del problema 9.29

Chiamiamo 3 COLORABILITY OR COMPLETE GRAPH (in breve, 3COLC) il problema in questione, che può essere formalizzato come di seguito descritto:

- $I_{3COLC} = \{G = (V,E) : G \text{ è un grafo non orientato } \}$;
- $S_{3COLC}(G) = \{c : V \rightarrow \{1,2,3\} \}$;
- $\pi_{3COLC}(G, S_{3COLC}(G)) = \left\{ \exists c \in S_{3COLC}(G) : \ \forall (u,v) \in E \ [\ c(u) \neq c(v) \ ] \ \right\} \ \vee \ G$ è un grafo completo.

Il problema è in **NP**: infatti, un certificato per una istanza $G = (V,E)$ è una coppia $\langle c, G \rangle$, in cui $c : V \rightarrow \{1,2,3\}$ è una colorazione dei nodi dei G mediante 3 colori, che ha lunghezza $\mathbf{O}(|G|)$, e verificare se un certificato è una soluzione effettiva significa verificare il predicato

$$\eta_{3COLC}(G,c) = \forall (u,v) \in E \ [\ c(u) \neq c(v) \ ] \ \vee \ \forall u,v \in V \ [\ (u,v) \in E \ ],$$

verifica, quest'ultima, realizzabile in tempo $\mathbf{O}(|E||V|^2)$.

Inoltre, il problema è completo per **NP**. Per dimostrarlo, mostriamo una riduzione polinomiale f dal problema 3COL. Prima di procedere, osserviamo che ogni istanza di 3COL costituita da un grafo contenente uno o due nodi è, banalmente, un'istanza no. In altre parole, il problema 3COL ristretto a grafi di uno o due nodi è un problema in **P**. Di conseguenza, poiché 3COL è un problema **NP**-completo in generale, esso rimane **NP**-completo per grafi che contengono almeno 3 nodi. In conclusione, nella riduzione che stiamo per presentare assumeremo, senza perdita di generalità, che il grafo istanza di 3COL contenga almeno 3 nodi.
Presentiamo, ora, la riduzione: data una istanza di 3COL $G = (V,E)$ (con $|V| \geq 3$), la corrispondente istanza di 3COLC è $f(G) = G' = (V', E')$, dove

- $V' = V \cup \{a\}$, in cui a è un nuovo nodo (ossia, $a \notin V$),
- $E' = E$.

Informalmente, il grafo G' è ottenuto semplicemente aggiungendo un nodo isolato a G.
Banalmente, calcolare G' da G richiede tempo lineare in $|G|$.

Mostriamo ora che f è effettivamente una riduzione da 3COL a 3COLC. Allo scopo, osserviamo che ogni 3-colorazione di G è banalmente trasformabile in una 3-colorazione di G' (assegnando ad a uno qualsiasi dei 3 colori): dunque, se G è una istanza sì di 3COL, allora G' è una istanza sì di 3COLC. Viceversa, anche ogni 3-colorazione di G' è banalmente trasformabile in una 3-colorazione di G e, quindi, se G' è una istanza sì di 3COLC, allora G è una istanza sì di 3COL.

Questo completa la prova di **NP**-completezza del problema.

Soluzione del problema 9.30

Chiamiamo 3-COLORABILITY OR CLIQUE (in breve, 3COL$\vee$C) il problema in questione, che può essere formalizzato come di seguito descritto:

- $I_{3COL\vee C} = \{\langle G = (V,E), k \rangle : G \text{ è un grafo non orientato e } k \in \mathbb{N}\}$;
- $S_{3COL\vee C}(G,k) = \{\langle c, V' \rangle : \ c : V \rightarrow \{1,2,3\} \ \wedge \ V' \subseteq V \ \}$;
- $\pi_{3COL\vee C}(G,k,S_{3COL\vee C}(G,k)) = \exists \langle c, V' \rangle \in S_{3COL\vee C}(G,k) : \ \left\{ \ \forall (u,v) \in E \ [\ c(u) \neq c(v) \ ] \ \right\} \ \vee \ \left\{ \ |V'| \geq |V| - k \ \wedge \ \forall \ u,v \in V' \ [\ (u,v) \in E \ ] \ \right\}$.

Il problema è in **NP**: infatti, un certificato per una istanza $G = (V,E)$ è una coppia $\langle c, V' \rangle$, in cui $c : V \rightarrow \{1,2,3\}$ è una colorazione dei nodi dei G mediante 3 colori, che ha lunghezza $\mathbf{O}(|G|)$, e V' è un sottoinsieme di V, che ha lunghezza $\mathbf{O}(|V|)$. Inoltre, verificare se un certificato è una soluzione effettiva significa verificare il predicato

$$\eta_{3COL\vee C}(G,k,c,V') = \left\{ \ \forall (u,v) \in E \ [\ c(u) \neq c(v) \ ] \ \right\} \ \vee \ \left\{ \ |V'| \geq |V| - k \ \wedge \ \forall u,v \in V \ [\ (u,v) \in E \ ] \ \right\},$$

verifica, quest'ultima, realizzabile in tempo $\mathbf{O}(|E||V|^2)$.

Inoltre, il problema è completo per **NP**. Per dimostrarlo, mostriamo una riduzione polinomiale f dal problema 3-COLORABILITÀ (3COL): data una istanza $G = (V,E)$ di 3COL, la corrispondente istanza di 3COL∨C è $f(G) = \langle \overline{G} = (\overline{V}, \overline{E}), k \rangle$, dove

- $\overline{V} = V \cup \{a\}$, ove $a \notin V$,
- $\overline{E} = E$,
- $k = 0$.

Informalmente, il grafo $\overline{G}$ è ottenuto semplicemente aggiungendo un nuovo nodo isolato a G: avendo scelto $k = 0$, garantiamo in questo modo che, qualunque sia G, il grafo $\overline{G}$ non è un grafo completo, o, in altri termini, $\overline{G}$ non contiene un sottografo completo di $|\overline{V}| - k = |\overline{V}|$ nodi.
Banalmente, calcolare $\overline{G}$ da G richiede tempo lineare in $|G|$.

Mostriamo ora che f è effettivamente una riduzione da 3COL a 3COL∨C.
Sia $G = (V,E)$ una istanza sì di 3COL; allora, esiste $c : V \rightarrow \{1,2,3\}$ tale che, per ogni $(u,v) \in E$, $c(u) \neq c(v)$. Consideriamo, allora, la seguente funzione $\overline{c} : \overline{V} \rightarrow \{1,2,3\}$: per ogni $\overline{u} \in \overline{V}$,

$$\overline{c}(\overline{u}) = \begin{cases} 1 & \text{se } \overline{u} = a, \\ c(\overline{u}) & \text{se } \overline{u} \neq a. \end{cases}$$

Banalmente, poiché c è una 3-colorazione valida per G, anche $\overline{c}$ è una 3-colorazione valida per $\overline{G}$. Questo prova che $\langle \overline{G}, k \langle$ è una istanza sì di 3COL∨C.
Sia $\langle G = (V,E) \rangle$ una istanza no di 3COL e supponiamo che $\langle \overline{G}, k = 0 \rangle$, l'istanza di 3COL∨C corrispondente a G, sia invece una istanza sì per 3COL∨C. Poiché, come abbiamo già osservato, $\overline{G}$ non contiene un sottografo completo di $|V| - k$ nodi, affinché $\langle \overline{G}, k = 0 \rangle$ sia una istanza sì per 3COL∨C deve accadere che $\overline{G}$ sia 3-colorabile: dunque, deve esistere una funzione $\overline{c} : \overline{V} \rightarrow \{1,2,3\}$ tale che, per ogni $(\overline{u}, \overline{v}) \in \overline{E}$, sia $\overline{c}(\overline{u} \neq \overline{c}(\overline{v})$. Definiamo, allora, la seguente funzione $c : V \rightarrow \{1,2,3\}$:

$$\text{per ogni } u \in V,\ c(u) = c(u).$$

Poiché $\overline{E} = E$, e poiché $\overline{c}$ è una 3-colorazione per $\overline{G}$, per ogni $(u,v) \in E$, si ha che $c(u) \neq c(v)$. Ma questo significa che c è una 3-colorazione valida per G, contraddicendo, in tal modo, il fatto che G non è 3-colorabile. Quindi, $\overline{G}$ è una istanza no di 3COL∨C.

Questo completa la prova di **NP**-completezza del problema.

Soluzione del problema 9.31

Il problema in esame, che indicheremo, in breve, con l'acronimo MaxTrue3SAT, può essere formalizzato come di seguito descritto:

- $I_{\texttt{MaxTrue3SAT}} = \{ \langle X, f, k \rangle : f \text{ è una funzione booleana in 3CNF nelle variabili in } X \text{ e } k \in \mathbb{N} \}$;
- $S_{\texttt{MaxTrue3SAT}}(X, f, k) = \{ a : X \rightarrow \{\texttt{vero}, \texttt{falso}\} \}$;
- $\pi_{\texttt{MaxTrue3SAT}}(X, f, k, S_{\texttt{MaxTrue3SAT}}(X, f, k)) = \exists a \in S_{\texttt{MaxTrue3SAT}}(X, f, k) : \ f(a(X)) \ \wedge \ |\ \{ \ x \in X : a(x) = \texttt{vero} \ \} \ | \leq k$.

Un certificato per una istanza $\langle X, f, k \rangle$ di MaxTrue3SAT è un elemento $a \in S_{\texttt{MaxTrue3SAT}}(X, f, k)$ e, dunque ha lunghezza $\mathbf{O}(|X|)$. Per verificare un certificato è necessario verificare che esso soddisfa f (e sappiamo dal problema 3SAT che tale verifica richiede tempo polinomiale in $|f|$ e $|X|$) e che $|\ \{ \ x \in X : a(x) = \texttt{vero} \ \} \ | \leq k$: banalmente, quest'ultimo test richiede tempo lineare in $|X|$ e $|k|$. Questo prova che il problema è in **NP**.

Dimostriamo, ora, la completezza del problema per la classe **NP** mediante una riduzione polinomiale dal problema 3SAT. Sia $\langle X, f \rangle$ una istanza di 3SAT: ad essa facciamo corrispondere l'istanza $\langle X, f, |X| \rangle$ di MaxTrue3SAT, che, banalmente, è calcolabile in tempo polinomiale da $\langle X, f \rangle$.

Se $\langle X, f \rangle$ è una istanza sì di 3SAT, allora esiste una assegnazione di verità per X tale che $f(a(X)) = \texttt{vero}$. Ovviamente, $|\ \{ x \in X : a(x) = \texttt{vero} \}\ | \leq |X| = k$, e, quindi, $\langle X, f, |X| \rangle$ è una istanza sì di MaxTrue3SAT.
Se, invece, $\langle X, f, |X| \rangle$ è una istanza sì di MaxTrue3SAT, allora esiste una assegnazione di verità per X tale che $f(a(X)) = \texttt{vero}$ e $|\ \{ x \in X : a(x) = \texttt{vero} \}\ | \leq |X| = k$: tale assegnazione, dunque, soddisfa f e, quindi, $\langle X, f \rangle$ è una istanza sì di 3SAT.
Questo completa la prova di **NP**-completezza di MaxTrue3SAT.

Soluzione del problema 9.32

Il problema in esame, che indicheremo, in breve, con l'acronimo MinTrue3SAT, può essere formalizzato come di seguito descritto:

- $I_{\texttt{MinTrue3SAT}} = \{ \langle X, f, k \rangle : f$ è una funzione booleana in 3CNF nelle variabili in X e $k \in \mathbb{N}^+ \}$;
- $S_{\texttt{MinTrue3SAT}}(X, f, k) = \{ a : X \rightarrow \{\texttt{vero}, \texttt{falso}\} \}$;
- $\pi_{\texttt{MinTrue3SAT}}(X, f, k, S_{\texttt{MinTrue3SAT}}(X, f, k)) = \exists a \in S_{\texttt{MinTrue3SAT}}(X, f, k) :\ f(a(X)) \ \wedge \ |\ \{ x \in X : a(x) = \texttt{vero} \}\ | \geq k$.

Un certificato per una istanza $\langle X, f, k \rangle$ di MinTrue3SAT è un elemento $a \in S_{\texttt{MinTrue3SAT}}(X, f, k)$ e, dunque ha lunghezza $\mathbf{O}(|X|)$. Per verificare un certificato è necessario verificare che esso soddisfa f (e sappiamo dal problema 3SAT che tale verifica richiede tempo polinomiale in $|f|$ e $|X|$) e che $|\ \{ x \in X : a(x) = \texttt{vero} \}\ | \geq k$: banalmente, quest'ultimo test richiede tempo lineare in $|X|$ e $|k|$. Questo prova che il problema è in **NP**.

Dimostriamo, ora, la completezza del problema per la classe **NP** mediante una riduzione polinomiale dal problema 3SAT. Prima di procedere, osserviamo che è richiesto che il valore k nell'istanza di MinTrue3SAT sia strettamente positivo.
Sia $\langle X, f \rangle$ una istanza di 3SAT e siano $y_1, y_2, y_3 \notin X$: ad essa facciamo corrispondere l'istanza $\langle X', f', 1 \rangle$ di MinTrue3SAT tale che: $X' = X \cup \{y_1, y_2, y_3\}$ e $f' = f \ \wedge \ (y_1 \vee y_2 \vee y_3)$. Banalmente, $\langle X', f', 1 \rangle$ è calcolabile in tempo polinomiale da $\langle X, f \rangle$.
Se $\langle X, f \rangle$ è una istanza sì di 3SAT, allora esiste una assegnazione di verità per X tale che $f(a(X)) = \texttt{vero}$. Consideriamo l'assegnazione $a' : X \cup \{y_1, y_2, y_3\}$ definita come segue:

$$a'(x) = \begin{cases} a(x) & \text{se } x \in X, \\ \textbf{vero} & \text{se } x = y_1, \\ \textbf{falso} & \text{se } x = y_2 \ \vee \ x = y_3. \end{cases}$$

Per costruzione, a' soddisfa f' e, inoltre, $|\ \{ x \in X' : a'(x) = \texttt{vero} \}\ | \geq 1 = k$, e, quindi, $\langle X', f', 1 \rangle$ è una istanza sì di MinTrue3SAT.
Se, invece, $\langle X', f', 1 \rangle$ è una istanza sì di MinTrue3SAT, allora esiste una assegnazione di verità a' per X' tale che $f'(a'(X')) = \texttt{vero}$: ancora per costruzione, la restrizione a di tale assegnazione alle variabili in X soddisfa f e, quindi, $\langle X, f \rangle$ è una istanza sì di 3SAT.
Questo completa la prova di **NP**-completezza di MinTrue3SAT.

Soluzione del problema 9.33

Il problema in esame, che indicheremo, in breve, con l'acronimo $3COL \vee kVC$, può essere formalizzato come di seguito descritto:

- $I_{3COL \vee kVC} = \{ G = (V, E) : G$ è un grafo non orientato $\}$;
- $S_{3COL \vee kVC}(G) = \{ \langle c, V' \rangle :\ c : V \rightarrow \{1, 2, 3\} \ \wedge \ V' \subseteq V \}$;
- $\pi_{3COL \vee kVC}(G, S_{3COL \vee kVC}(G)) = \exists \langle c, V' \rangle \in S_{3COL \vee kVC}(G) :\ \{ \forall (u, v) \in E \ [\ c(u) \neq c(v)\ ]\ \} \vee$
  $\{\ |V'| \leq k \wedge \forall (u, v) \in E \ [u \in V' \ \vee \ v \in V'\ ]\ \}$.

Un certificato per una istanza $\langle X, f, k\rangle$ di $3COL \vee kVC$ è una coppia $\langle c, V'\rangle \in S_{3COL\vee kVC}(G)$, e, dunque, poiché $|c| \in \mathbf{O}(|V|)$ e $|V'| \in \mathbf{O}(|V|)$, ha lunghezza $\mathbf{O}(|V|)$. Per verificare un certificato è necessario verificare che c sia una colorazione valida per G (ossia, che colori con colori diversi nodi adiacenti) o che V' sia un Vertex Cover di G di dimensione non superiore a k: poiché sia il problema COLORABILITÀ che il problema VERTEX COVER sono in **NP**, sappiamo che tale verifica richiede tempo polinomiale in $|V|$ e $|E|$). Questo prova che il problema è in **NP**.

Dimostriamo, ora, la completezza del problema per la classe **NP** mediante una riduzione polinomiale dal problema 3COL.
Prima di procedere, cerchiamo di capire perché riduciamo da 3COL invece che da VC. A questo scopo, osserviamo che, poiché k è costante (e, infatti, non è dichiarata nella definizione dell'insieme $I_{3COL\vee kVC}$) la versione di VERTEX COVER di interesse in questo problema è in **P**. Infatti, dato un grafo G, per verificare se G ha un Vertex Cover di k nodi è sufficiente verificare se uno degli $\mathbf{O}(|V|^k)$ sottoinsiemi di V contenenti k nodi sia un Vertex Cover per G: poiché la verifica richiede tempo polinomiale in $|V|$ e poicé k è costante, questo prova che il problema è in **P**. Quindi, ridurre da un problema in **P** non sarebbe di alcuna utilità per mostrare la **NP**-completezza di $3COL \vee kVC$.
Sia, dunque, G una istanza di 3COL; l'istanza corrispondente di $3COL \vee kVC$ è il grafo $\overline{G} = (\overline{V}, \overline{E})$ ottenuto aggiungendo a G $k+1$ nuovi grafi, che non hanno nodi in comune con G, ciascuno dei quali costituito da un singolo arco. Più in dettaglio: $\overline{G} = G \cup G_1 \cup G_2 \cup \ldots \cup G_{k+1}$, dove, per ogni $i = 1, \ldots, k+1$, $G_i = (V_i, E_i)$ e

- $V_i = \{u_i, v_i\}$, e
- $E_i = \{(u_i, v_i)\}$.

Dunque, $\overline{V} = V \cup V_1 \cup \ldots \cup V_{k+1}$ e $\overline{E} = V \cup E_1 \cup \ldots \cup E_{k+1}$.
Se $G = (V, E)$ è una istanza sì di 3COL, allora esiste una colorazione $c : V \to V$ dei nodi di G che non assegna lo stesso colore a due nodi adiacenti (ossia, collegati da un arco); allora la colorazione $\overline{c}$ tale che, per ogni $x \in \overline{V}$

$$\overline{c}(x) = \begin{cases} c(x) & \text{se } x \in V, \\ 1 & \text{se } x \in V_i \text{ e } x = u_i, \\ 2 & \text{se } x \in V_i \text{ e } x = v_i, \end{cases}$$

è una colorazione valida per $\overline{G}$, ossia assegna colori diversi a nodi adiacenti in $\overline{G}$.
Se, invece, $G = (V, e)$ è una istanza no di 3COL, allora non esiste alcuna colorazione di V con 3 colori che non assegni lo stesso colore ad almeno una coppia di nodi adiacenti in G; allora, poiché $\overline{G}$ contiene G, non esiste nemmeno alcuna colorazione di $\overline{V}$ con 3 colori che non assegni lo stesso colore ad almeno una coppia di nodi adiacenti in $\overline{G}$. D'altra parte, poiché ogni Vertex Cover di $\overline{G}$ deve almeno contenere un nodo in ciascun V_i, per $i = 1, \ldots, k+1$, ogni Vertex Cover di $\overline{G}$ ha cardinalità almeno $k+1$. Dunque, $\overline{G}$ è una istanza no di $3COL \vee kVC$.
Poiché costruire $\overline{G}$ a partire da G richiede tempo lineare in $|V|$ e $|E|$, questo dimostra che il problema $3COL \vee kVC$ è **NP**-completo.

Soluzione del problema 9.34

Il problema in esame, che indicheremo, in breve, con l'acronimo $2COL \vee CL$, può essere formalizzato come di seguito descritto:

- $I_{2COL\vee CL} = \{\langle G = (V, E), k\rangle : G \text{ è un grafo non orientato} \wedge k \in \mathbb{N} \wedge k \geq 3\}$;
- $S_{2COL\vee CL}(G) = \{\langle c, V'\rangle : \ c : V \to \{1, 2\} \ \wedge \ V' \subseteq V\} \}$;
- $\pi_{2COL\vee CL}(G, S_{2COL\vee CL}(G)) = \exists \langle c, V'\rangle \in S_{2COL\vee CL}(G) : \ \{\forall (u, v) \in E \ [\ c(u) \neq c(v)\ ]\ \} \vee$
  $\{\ |V'| \geq k \wedge \forall u, v \in V' \ [(u, v) \in E\ ]\ \}$.

Un certificato per una istanza $\langle G = (V, E), k\rangle$ di $2COL \vee CL$ è una coppia $\langle c, V'\rangle \in S_{2COL\vee CL}(G)$, e, dunque, poiché $|c| \in \mathbf{O}(|V|)$ e $|V'| \in \mathbf{O}(|V|)$, ha lunghezza $\mathbf{O}(|V|)$. Per verificare un certificato è necessario verificare che c sia una colorazione valida per G (ossia, che colori con colori diversi nodi adiacenti) o che V' sia una clique in G di dimensione non superiore a k: poiché sia il problema COLORABILITÀ che il problema CLIQUE sono in **NP**, sappiamo che tale verifica richiede tempo polinomiale in $|V|$ e $|E|$). Questo prova che il problema è in **NP**.

Dimostriamo, ora, la completezza del problema per la classe **NP** mediante una riduzione polinomiale dal problema CLIQUE.
Prima di procedere, cerchiamo di capire perché riduciamo da CLIQUE invece che da 2COL. A questo scopo, osserviamo che il problema 2COL è in **P** e ridurre da un problema in **P** non sarebbe di alcuna utilità per mostrare la **NP**-completezza di $2COL \vee CL$. Osserviamo, inoltre, che il problema CLIQUE ristretto ad istanze $\langle G = (V,E), k\langle$ tali che $k \geq 3$ rimane un problema **NP**-completo.
Sia, dunque, $\langle G = (V,E), k\rangle$ una istanza di CLIQUE tale che $k \geq 3$; l'istanza corrispondente di $2COL \vee CL$ è $\langle \overline{G} = (\overline{V}, \overline{E}), k\rangle$, dove il grafo $\overline{G} = (\overline{V}, \overline{E})$ è ottenuto aggiungendo a G un nuovo sottografo, costituito da un ciclo di 5 nuovi nodi, che non ha archi che lo collegano a G. Più in dettaglio:

- $\overline{V} = V \cup \{x_1, x_2, x_3, x_4, x_5 : x_1, x_2, x_3, x_4, x_5 \notin V\}$, e
- $\overline{E} = E \cup \{(x_1,x_2), (x_2,x_3), (x_3,x_4), (x_4,x_5), (x_5,x_1)\}$.

Se $\langle G = (V,E), k\rangle$ è una istanza sì di CLIQUE, allora esiste una colorazione $V' \subseteq V$ tale che $|V'| \geq k$ e ogni coppia di nodi di V' è collegata da un arco; allora V' gode delle stesse proprietà in $\overline{G}$ che, pertanto è una istanza sì di $2COL \vee CL$.
Se, invece, $\langle G = (V,E), k\rangle$ è una istanza no di CLIQUE, allora non esiste alcun sottografo completo di almeno k nodi in G; allora, poiché il grafo che abbiamo aggiunto a G per ottenere $\overline{G}$ èe un ciclo di 5 nodi e, dunque, non èe un grafo completo, allora anche $\overline{G}$ non contiene alcun sottografo completo di almeno k nodi. Allora, $\overline{G}$ è una istanza no di $2COL \vee CL$.
Poiché costruire $\overline{G}$ a partire da G richiede tempo lineare in $|V|$ e $|E|$, questo dimostra che il problema $2COL \vee CL$ è **NP**-completo.

2.3 coNP

Soluzione del problema 9.35

Formalizzazione del problema:

- $I = \{\langle G = (V,E), k\rangle : G$ è un grafo non orientato e k un intero positivo$\}$;
- osserviamo ora che per ciascuna istanza $\langle G, k\rangle$ di NO-CLIQUE esiste una *unica soluzione possibile*, ossia, *l'insieme di tutti i sottoinsiemi di V di cardinalità maggiore o uguale a k*: detto $\mathscr{V}_k = \{V' \subseteq V : |V'| \geq k\}$, allora $S(G,k) = \{\mathscr{V}_k\}$;
- $\pi(G, k, (V)_k) = \forall V' \in \mathscr{V}_k \exists u, v \in V' : (u,v) \notin E$.

Osserviamo subito che la dimensione della (unica) soluzione possibile di una istanza di NO-CLIQUE è $O(|V|^k)$: sembra, dunque, poco probabile riuscire a verificare il predicato π in tempo polinomiale in $|V|$ e k. Questa osservazione dovrebbe indurre a ritenere poco probabile che il problema in questione appartenga ala classe NP.

Passiamo ora all'analisi formale. Consideriamo ora una istanza $\langle G, k\rangle \in I$ che non appartenga al linguaggio NO-CLIQUE: allora, V contiene un sottoinsieme V' di almeno k nodi tale che ogni coppia di nodi di V' è collegata da un arco. Questo significa che V' è una clique in G di almeno k nodi. Questo dimostra che l'insieme delle istanze $\langle G, k\rangle \in I$ che *non* appartengono a NO-CLIQUE coicide con l'insieme delle istanze che appartengono a CLIQUE, ossia NO-CLIQUE=CLIQUE^c. Dunque, NO-CLIQUE$\in$Co-NP e, in particolare, poiché CLIQUE è completo rispetto ad NP, NO-CLIQUE è Co-NP-completo.

Soluzione del problema 9.36

Il problema in esame, che chiameremo T, è formalizzato nella maniera seguente:

- $I_T = \{\langle G = (V,E), k\rangle : G$ è un grafo non orientato e k un intero positivo$\}$;

- osserviamo ora che per ciascuna istanza $\langle G,k \rangle$ di T esiste una *unica soluzione possibile*, ossia, *l'insieme di tutti i sottoinsiemi di V di cardinalità maggiore o uguale a k*: detto $\mathscr{P}(V,k) = \{V' \subseteq V : |V'| \geq k\}$, allora $S_T(G,k) = \{\mathscr{P}(V,k)\}$;
- $\pi_T(G,k,(P)(V,k)) = \forall V' \in \mathscr{P}(V,k) \exists u,v \in V' : (u,v) \in E = \forall V' \subseteq V : |V'| \geq k[\exists u,v \in V' : (u,v) \in E]$.

Osserviamo subito che la dimensione della (unica) soluzione possibile di una istanza di T è $O(|V|^k)$: sembra, dunque, poco probabile riuscire a generare tale soluzione in tempo non deterministico polinomiale, in quanto per generarla occorre almeno enumerare tutti i suoi elementi. Sembra, quindi, poco probabile che il problema in questione appartenga alla classe **NP**.

Passiamo ora all'analisi formale. Consideriamo ora una istanza $\langle G,k \rangle \in I_T$. Se $\langle G,k \rangle \notin$T, allora esiste un sottoinsieme V' di V di almeno k che contiene una coppia di nodi collegati da un arco: quindi, G contiene un insieme indipendente di cardinalità almeno k, ossia, $\langle G,k \rangle \in$INSIEME INDIPENDENTE. Viceversa, se $\langle G,k \rangle \in$T, allora G non contiene alcun insieme indipendente di almeno k nodi, ossia, $\langle G,k \rangle \notin$INSIEME INDIPENDENTE. In conclusione, T = (INSIEME INDIPENDENTE)^c. Dunque, T∈Co-**NP**e, in particolare, poiché INSIEME INDIPENDENTE è completo rispetto ad **NP**, T è Co-**NP**-completo.

Soluzione del problema 9.37

Si osservi che, dati un grafo non orientato $G = (V,E)$ e un intero positivo k, il problema consiste nel decidere se G non contiene alcuna clique di $k+1$ nodi. Il problema coincide, dunque, con il complemento del problema CLIQUE. Conseguentemente, esso è co**NP**-completo.

Soluzione del problema 9.38

Si consideri il problema decisionale seguente: dato un grafo non orientato $G = (V,E)$, decidere se ogni ciclo in G ha lunghezza $< |V| - 1$. Studiare la complessità computazionale di tale problema.
Si ricordi che un ciclo di lunghezza k in un grafo non orientato $G = (V,E)$ è una sequenza $\langle v_1, v_2, \dots v_k \rangle$ di elementi distinti di V tali che, per $i = 1, \dots k-1$, $(v_i, v_{i+1}) \in E$ e, inoltre, $(v_k, v_1) \in E$.

Indichiamo con NOLONGCYCLE (in breve, *NLC*) il problema in questione e consideriamo il suo complemento NLC^c: decidere se un dato grafo $G = (V,E)$ contiene almeno un ciclo di lunghezza $|V| - 1|$. Il problema NLC^c può essere deciso mediante una serie di invocazioni dell'algoritmo non deterministico che decide il problema CIRCUITO HAMILTINIANO, come illustrato di seguito:

Input: $G = (V,E)$, con $V = \{v_1, v_2, \dots, v_n\}$.

1) $i \leftarrow 1$;
2) $trovato \leftarrow falso$;
3) **while** ($i \leq n \wedge trovato = falso$) **do begin**

   (si sceglie quale nodo non considerare per costruire un ciclo di $n-1$ nodi)

   3.1) **if** (il grafo $G_i = (V - \{v_i\}, E_i)$ dove $E_i = E - \{(v, v_i) : v \in V\}$ contiene un ciclo hamiltoniano) **then**

   3.1.1) $esito \leftarrow vero$;

   end
4) **if** (G contiene un ciclo hamiltoniano) **then** $esito \leftarrow vero$;
5) **Output:** $esito$.

Il precedente algoritmo decide NLC^c ed opera in tempo non deterministico polinomiale. Dunque, NLC^c è in **NP** e NLC è in co**NP**.
Infine, il problema NLC^c è completo per **NP**. Per dimostrare questa affermazione consideriamo la seguente riduzione f da CIRCUITO HAMILTONIANO a NLC^c: dato un grafo non orientato $G = (V, E)$ (istanza di CIRCUITO HAMILTONIANO, $G' = f(G)$ è il grafo ottenuto aggiungendo a G un nodo isolato. Banalmente, f è in **FP** e, inoltre, G è una istanza sì di CIRCUITO HAMILTONIANO se e soltanto se G' è una istanza sì di NLC^c. Quindi, NLC^c è **NP**-completo e, in conclusione, NLC è co**NP**-completo.

Soluzione del problema 9.39

Sia $G = (V, E)$ un grafo non orientato e $k \in \mathbb{N}$. La coppia $\langle G, k \rangle$ è una istanza no del problema in esame se e soltanto se

esiste in G una coppia di nodi $u, v \in V$ tali che G contiene un percorso da u a v di lunghezza almeno k.

Ossia, $\langle G, k \rangle$ è una istanza no del problema in esame se e soltanto se $\langle G, k \rangle$ è una istanza sì del problema LONGEST PATH. Dunque, il problema in esame è il complemento di LONGEST PATH. Dalla **NP**-completezza del problema LONGEST PATH segue che il problema in esame è Co**NP**-completo.

Soluzione del problema 9.40

Il problema GNI può essere formalizzato come di seguito descritto:

- $I_{GNI} = \{\langle G = (V, E), G' = (V', E') \rangle : G \text{ e } G' \text{ sono grafi non orientati } \}$;
- $S_{GNI}(G, G') = \{b : V \to V' : \ b \text{ è una biezione fra } V \text{ e } V' \}$;
- $\pi_{GNI}(G, G', S_{GNI}(G, G')) = \forall b \in S_{GNI}(G, G') \ \left[\ \exists u, v \in V : \ \big((u, v) \in E \wedge (b(u), b(v)) \notin E \big) \ \vee \ \big((u, v) \notin E \wedge (b(u), b(v)) \in E \big) \ \right]$.

Osserviamo che il predicato π_{GNI} del problema richiede di verificare che *ogni* soluzione possibile soddisfi la proprietà:

$$\exists u, v \in v : \ \big((u, v) \in E \wedge (b(u), b(v)) \notin E \big) \ \vee \ \big((u, v) \notin E \wedge (b(u), b(v)) \in E \big).$$

La necessità di eseguire questa verifica per ogni elemento di $S_{GNI}(G, G')$ suggerisce che il problema sia in co**NP**. Per dimostrarlo, consideriamo il problema complemento, GRAFI ISOMORFI (GI): dati due grafi non orientati $G = (V, E)$ e $G' = (V', E')$, decidere se G e G' sono isomorfi. Di seguito, la sua formalizzazione:

- $I_{GI} = \{\langle G = (V, E), G' = (V', E') \rangle : G \text{ e } G' \text{ sono grafi non orientati } \}$;
- $S_{GI}(G, G') = \{b : V \to V' : \ b \text{ è una biezione fra } V \text{ e } V' \}$;
- $\pi_{GI}(G, G', S_{GNI}(G, G')) = \exists b \in S_{GNI}(G, G') : \ \forall u, v \in V \ \left[\ (u, v) \in E \Leftrightarrow (b(u), b(v)) \in E \ \right]$.

Il problema GI è in **NP**: infatti, un certificato per una istanza $\langle G = (V, E), G' = (V', E') \rangle$ è una funzione $b : V \to V'$, che ha lunghezza $\mathbf{O}(|V|)$. Inoltre, verificare se un certificato è una soluzione effettiva significa verificare il predicato

$$\eta_{GI}(G, G', b) = \forall u, v \in V \ \left[\ (u, v) \in E \Leftrightarrow (b(u), b(v)) \in E \ \right],$$

verifica, quest'ultima, realizzabile in tempo $\mathbf{O}(|V|^2 |E'|)$. Questo completa la dimostrazione che GI è in **NP** e, conseguentemente, che il suo complemento GNI è in co**NP**.

Soluzione del problema 9.41

Indicheremo i due problemi in esame, rispettivamente, con l'acronimo Γ_a e Γ_b.
Entrambi i problemi sono definiti sull'insieme di istanze I_Γ e sull'insieme di soluzioni possibili $_\Gamma$ di seguito descritti:

- $I_\Gamma = \{\langle G=(V,E),k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}^+\}$;
- $S_\Gamma(G,k) = \{V' : V' \subseteq V\}$ }.

I predicati dei due problemi sono, invece, distinti.

Il predicato π_{Γ_a} del problema Γ_a è molto simile al predicato che definisce il problema VERTEX COVER, e differisce da quest'ultimo soltanto per la cardinalità del vertex cover richiesto:

$$\pi_{\Gamma_a}(G,k,S_\Gamma(G,k)) = \exists\, V' \in S_\Gamma(G,k) :\ |V'| > k \ \wedge\ \forall\, (u,v) \in E\ [\ u \in V' \ \vee\ v \in V'\ ].$$

Poiché ogni grafo $G=(V,E)$ ha, banalmente, un vertex cover di $|V!$ nodi (e, altrettanto banalmente, non ha un vertex cover con più di $|V|$ nodi), per decidere se una istanza $\langle G=(V,E),k\rangle$ di Γ_a è una istanza sì è sufficiente verificare se $k < |V|$: in caso affermativo $\langle G=(V,E),k\rangle$ è una istanza sì, in caso negativo $\langle G=(V,E),k\rangle$ è una istanza no. Questo prova che il problema Γ_a è in **P**.

Il predicato π_{Γ_b} del problema Γ_b, pur essendo collegato al predicato di VERTEX COVER, è sostanzialmente diverso da esso in quanto richiede che *ogni* soluzione possibile soddisfi una certa proprietà. In particolare, π_{Γ_b} richiede che, se una soluzione possibile è un vertex cover, allora la sua cardinalità deve essere maggiore di k:

$$\pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \forall\, V' \in S_\Gamma(G,k) :\ \big[\ \forall\, (u,v) \in E\ [\ u \in V' \ \vee\ v \in V'\ ]\ \rightarrow |V'| > k\ \big],$$

che può essere anche scritto

$$\pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \forall\, V' \in S_\Gamma(G,k) :\ \big[\ \neg\big(\forall\, (u,v) \in E\ [\ u \in V' \ \vee\ v \in V'\ ]\ \big)\ \vee\ |V'| > k\ \big].$$

Consideriamo ora la negazione del predicato π_{Γ_b}:

$$\neg\ [\pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \exists\, V' \in S_\Gamma(G,k) :\ \forall\, (u,v) \in E\ [\ u \in V' \ \vee\ v \in V'\ ]\ \wedge\ |V'| \leq k.$$

Osserviamo, ora, che detti I_{VC}, S_{VC} e π_{VC}, rispettivamente, l'insieme delle istanze, l'insieme delle soluzioni possibili e il predicato del problema VERTEX COVER, si ha che $I_{VC} = I_\Gamma$ e, per ogni $\langle G=(V,E),k\rangle \in I_\Gamma$, $S_{VC}(G,k) = S_\Gamma(G,k)$ e $\neg\ [\ \pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \pi_{VC}(G,k,S_{VC}(G,k))$. Questo signifca che il problema Γ_b^c, complemento di Γ_b, coincide con il problema VERTEX COVER. Quindi, Γ_b^c è **NP**-completo e Γ_b è co**NP**-completo.

Soluzione del problema 9.42

Indicheremo i due problemi in esame, rispettivamente, con l'acronimo Γ_a e Γ_b.
Entrambi i problemi sono definiti sull'insieme di istanze I_Γ e sull'insieme di soluzioni possibili $_\Gamma$ di seguito descritti:

- $I_\Gamma = \{\langle G=(V,E),k\rangle : G$ è un grafo non orientato e $k \in \mathbb{N}^+\}$;
- $S_\Gamma(G,k) = \{V' : V' \subseteq V\}$ }.

I predicati dei due problemi sono, invece, distinti.

Il predicato π_{Γ_a} del problema Γ_a è molto simile al predicato che definisce il problema CLIQUE, e differisce da quest'ultimo soltanto per la cardinalità del sottografo completo richiesto:

$$\pi_{\Gamma_a}(G,k,S_\Gamma(G,k)) = \exists\, V' \in S_\Gamma(G,k) :\ |V'| < k \ \wedge\ \forall\, u,v \in V'\ [\ (u,v) \in E\ ].$$

Osserviamo che, nel predicato del problema (così come in quello del problema CLIQUE), viene assunto implicitamente $u \neq v$, ossia, il predicato richiede che esista una soluzione possibile in cui ogni coppia di nodi *distinti* sia collegata da un arco. Esso, più propriamente, dovrebbe essere scritto nel modo seguente:

$$\exists\, V' \in S_\Gamma(G,k) :\ |V'| < k \ \wedge\ \forall\, u,v \in V' :\ u \neq v\ [\ (u,v) \in E\ ].$$

Pertanto, ciascun insieme V' contenente un solo nodo soddisfa banalmente il predicato, in quanto non contiene una coppia di nodi distinti. In altre parole, un singolo nodo di G è un sottografo completo di di dimensione 1.
Analogamente, l'insieme vuoto soddisfa banalmente il predicato in quanto non contiene alcun nodo. Ossia, l'insieme vuoto è un sottografo completo di G di dimensione 0. Quindi, per decidere se una istanza $\langle G=(V,E),k\rangle$ di Γ_a è una istanza sì è sufficiente verificare se $k>0$: in caso affermativo $\langle G=(V,E),k\rangle$ è una istanza sì, in caso negativo $\langle G=(V,E),k\rangle$ è una istanza no. Questo prova che il problema Γ_a è in **P**.

Il predicato π_{Γ_b} del problema Γ_b, pur essendo collegato al predicato di CLIQUE, è sostanzialmente diverso da esso in quanto richiede che *ogni* soluzione possibile soddisfi una certa proprietà. In particolare, π_{Γ_b} richiede che, se una soluzione possibile è un sottografo completo, allora la sua cardinalità deve essere minore di k:

$$\pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \forall\, V' \in S_\Gamma(G,k)\ \big[\ \forall\, u,v \in V'\ [\ (u,v) \in E\ ]\ \rightarrow |V'| < k\ \big],$$

che può essere anche scritto

$$\pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \forall\, V' \in S_\Gamma(G,k)\ \big[\ \neg\big(\forall\, u,v \in V'\ [\ (u,v) \in E\ ]\ \big)\ \vee\ |V'| < k\ \big].$$

Consideriamo ora la negazione del predicato π_{Γ_b}:

$$\neg\,[\pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \exists\, V' \in S_\Gamma(G,k) :\ \forall\, u,v \in V'\ [\ (u,v) \in E\ ]\ \wedge\ |V'| \geq k.$$

Osserviamo, ora, che detti I_{Cl}, S_{Cl} e π_{Cl}, rispettivamente, l'insieme delle istanze, l'insieme delle soluzioni possibili e il predicato del problema CLIQUE, si ha che $I_{Cl} = I_\Gamma$ e, per ogni $\langle G=(V,E),k\rangle \in I_\Gamma$, $S_{Cl}(G,k) = S_\Gamma(G,k)$ e $\neg\,[\ \pi_{\Gamma_b}(G,k,S_\Gamma(G,k)) = \pi_{Cl}(G,k,S_{VC}(G,k))$. Questo significa che il problema Γ_b^c, complemento di Γ_b, coincide con il problema CLIQUE. Quindi, Γ_b^c è **NP**-completo e Γ_b è co**NP**-completo.