Soluzioni Esercizi Automi

Gabriel Rovesti

June 2024

1 Problemi NP Completi

1.1 Esercizio 1

Parte 1: Dimostrare che il problema del circuito quasi Hamiltoniano è in NP

Il problema del circuito quasi Hamiltoniano in un grafo G richiede di trovare un ciclo che attraversa tutti i vertici del grafo esattamente una volta tranne uno. Per dimostrare che il problema è in NP, consideriamo il seguente approccio:

- Dato un certificato, ovvero un ciclo proposto che include tutti i vertici tranne uno, possiamo verificare in tempo polinomiale se ogni vertice appare esattamente una volta nel ciclo e se ogni coppia di vertici consecutivi nel ciclo è collegata da un arco nel grafo.
- È necessario anche verificare che il vertice escluso dal ciclo non sia collegato ad alcun vertice nel ciclo, assicurando che la soluzione proposta sia valida.

Questa verifica può essere realizzata in tempo polinomiale rispetto al numero di vertici e archi nel grafo, confermando che il problema è in NP.

Parte 2: Dimostrare che il problema del circuito quasi Hamiltoniano è NP-hard

Per dimostrare che il problema del circuito quasi Hamiltoniano è NP-hard, usiamo il problema del circuito Hamiltoniano, già noto per essere NP-hard, come riferimento per una riduzione polinomiale:

1. Partiamo da un'istanza del problema del circuito Hamiltoniano, definito su un grafo G.
2. Costruiamo un nuovo grafo G' aggiungendo un vertice aggiuntivo v a G e collegandolo con tutti gli altri vertici di G.
3. Ora, se il grafo G' contiene un circuito quasi Hamiltoniano che omette il vertice v, allora il grafo originale G possiede un circuito Hamiltoniano. Infatti, il ciclo che include tutti i vertici di G tranne v in G' corrisponde a un ciclo Hamiltoniano in G.

La riduzione da G a G' è computazionalmente efficiente e dimostra che il problema del circuito quasi Hamiltoniano è NP-hard in quanto ogni problema NP (nello specifico il circuito Hamiltoniano) può essere ridotto a esso in tempo polinomiale.

1.2 Esercizio 2

Parte 1: Dimostrare che SubsetSum è in NP

Il problema SubsetSum è definito come segue: dato un insieme S di numeri interi e un intero t, si chiede se esiste un sottoinsieme $S' \subseteq S$ tale che la somma dei numeri in S' sia esattamente t. Per verificare che SubsetSum è in NP, consideriamo un certificato S', che è un sottoinsieme candidato di S. La verifica consiste nel sommare gli elementi di S' e confrontare questa somma con t. Questa operazione può essere eseguita in tempo polinomiale rispetto alla dimensione dell'insieme S, quindi SubsetSum è in NP.

Parte 2: Dimostrare che SubsetSum è NP-Hard

Per dimostrare che SubsetSum è NP-hard, utilizziamo una riduzione dal problema SetPartitioning, noto essere NP-completo. La riduzione può essere descritta come segue:

1. Consideriamo un'istanza del problema SetPartitioning, dove l'obiettivo è dividere un insieme S di numeri interi in due sottoinsiemi $S1$ e $S2$ tali che la somma dei numeri in $S1$ sia uguale alla somma dei numeri in $S2$.

2. Definiamo t come la metà della somma totale degli elementi di S, assumendo che la somma totale sia pari. Se la somma totale degli elementi di S è dispari, il problema SetPartitioning non ha soluzione e quindi non è necessaria ulteriore considerazione.

3. Creiamo una nuova istanza di SubsetSum con il set S e il target t. Se esiste un sottoinsieme $S' \subseteq S$ che somma a t, allora S può essere diviso in $S1 = S'$ e $S2 = S \setminus S'$ con somme uguali.

Questo processo di riduzione è eseguibile in tempo polinomiale e dimostra che ogni istanza di SetPartitioning può essere trasformata in un'istanza di SubsetSum. Quindi, se SubsetSum è risolvibile, allora lo è anche SetPartitioning, mostrando che SubsetSum è NP-hard.

Esercizio 3

Parte 1: Mostrare che il problema k-COLOR è in NP

Il problema k-COLOR richiede di determinare se è possibile colorare i vertici di un grafo non orientato usando al massimo k colori, in modo tale che nessuna coppia di vertici adiacenti abbia lo stesso colore. Per dimostrare che il problema k-COLOR è in NP:

- Dato un certificato che consiste in una colorazione dei vertici del grafo, possiamo verificare in tempo polinomiale se la colorazione assegna ai vertici adiacenti colori diversi.
- La verifica che ogni vertice abbia uno dei k colori e che vertici adiacenti non condividano lo stesso colore può essere completata in tempo polinomiale rispetto al numero di vertici e archi del grafo.

Parte 2: Mostrare che 3-COLOR $\leq_p$ k-COLOR per ogni k ¿ 3

Per dimostrare che 3-COLOR si riduce polinomialmente a k-COLOR per ogni $k > 3$, consideriamo il seguente approccio:

1. Prendiamo un'istanza del problema 3-COLOR, che chiede se è possibile colorare i vertici di un grafo usando al massimo tre colori.
2. Poiché $k > 3$, ogni soluzione valida al problema 3-COLOR è anche una soluzione valida al problema k-COLOR, poiché se un grafo può essere colorato con 3 colori, può certamente essere colorato con k colori.
3. Non è necessaria alcuna modifica al grafo o alla colorazione; la stessa istanza può essere usata per dimostrare che se esiste una soluzione per 3-COLOR, esiste anche per k-COLOR.

Questa riduzione dimostra che risolvere 3-COLOR risolve anche k-COLOR quando $k \geq 3$, stabilendo che 3-COLOR $\leq_p$ k-COLOR.

1.3 Esercizio 4

Dimostra che PebbleDestruction è NP-hard usando il problema del Circuito Hamiltoniano

Il problema PebbleDestruction richiede di determinare se, dato un grafo $G = (V, E)$ con un numero di ciottoli $p(v)$ per ogni vertice v, è possibile eseguire una sequenza di mosse che rimuovano tutti i ciottoli tranne uno. Per dimostrare che PebbleDestruction è NP-hard, usiamo una riduzione dal problema del circuito Hamiltoniano, noto per essere NP-hard:

1. Consideriamo un'istanza del problema del circuito Hamiltoniano su un grafo G.
2. Costruiamo un'istanza del problema PebbleDestruction nel seguente modo: per ogni vertice v in G, assegniamo due ciottoli.
3. Definiamo le mosse possibili in modo che due ciottoli possano essere spostati da un vertice v a un vertice u se e solo se esiste un arco (v, u) in G.
4. Se G ha un circuito Hamiltoniano, possiamo sequenziare le mosse in modo che ogni vertice "trasferisca" un ciottolo al prossimo vertice nel circuito, lasciando alla fine un solo ciottolo.

Questa costruzione mostra che se esiste una soluzione al problema del circuito Hamiltoniano in G, allora esiste una soluzione al problema PebbleDestruction, dimostrando così che PebbleDestruction è NP-hard.