# Esame di Ricerca Operativa - 28 luglio 2016
# Facoltà di Scienze MM.FF.NN. - Verona
# - CORREZIONE -

**Problema 1 (3+2+2+4+1+1+1+1=15 punti):**

Per sostenere il lago con la cascata di cioccolato ai laboratori della Willy Wonka, gli Umpa Lumpa possono miscelare i seguenti tipi di cioccolato:

| Valori nutrizionali (per 1 Kg) | | | |
|---|---|---|---|
| | Tipologia di cioccolato | | |
| | fondente | al latte | bianco |
| Vitamina A (IU) | 400 | 3000 | 2200 |
| Vitamina B1 (mg) | 0,6 | 1 | 1 |
| Vitamina B2 (mg) | 0,6 | 3 | 4 |
| Vitamina C (mg) | 11,4 | 30 | 30 |
| Vitamina D (IU) | 500 | 700 | 150 |
| Vitamina E (mg) | 24 | 12 | - |

Il lago può essere rimpinguato fino ad un massimo di 240 volte al giorno, ma inserendo al più 10 Kg di cioccolato nuovo ad ogni afflusso per non alterare i delicati equilibri del suo biotopo; la prassi è pertanto quella di limitare i deflussi e di fare affluire precisamente 10 Kg di cioccolato ad ogni immissione. L'apporto minimo di vitamine richiesto ad ogni singola immissione è conseguentemente il seguente:

| Apporto minimo di vitamine ad ogni immissione | | | | | |
|---|---|---|---|---|---|
| Vitamina A | Vitamina B1 | Vitamina B2 | Vitamina C | Vitamina D | Vitamina E |
| 11000 (IU) | 8 (mg) | 16 (mg) | 134 (mg) | 3000 (IU) | 100 (mg) |

**(3pt)** Si formuli come un problema di PL il problema di minimizzare il costo di ogni singolo afflusso assumendo che il costo per i rabbocchi di cioccolato delle varie tipologie sia espresso dalla sola prima riga della seguente tabella:

| Costo (in euro) | | | |
|---|---|---|---|
| | Tipologia di cioccolato | | |
| | fondente | al latte | bianco |
| Costo per ogni Kg rabboccato | 70 | 80 | 60 |
| Costo attivazione tipologia | 120 | 130 | 101 |

**(2pt)** Si formuli come un problema di PLI il problema di minimizzare il costo di ogni singolo afflusso considerando che oltre ai costi lineari espressi dalla prima riga della tabella, come sopra, siano da considerarsi anche dei costi che scattano qualora si decida di avvalersi delle varie tipologie di cioccolato (cioccolato disperso nelle tubature da ripulire ogni volta). Questi costi di attivazione tipologia (costi di start-up per ogni singola tipologia) sono espressi nella seconda ed ultima riga della stessa tabella.

**(2=1+1pt)** Si effettui l'astrazione dai dati per entrambi i modelli, assumendo di dover miscelare i cioccolati $C_1, \ldots, C_n$ con costi lineari $CL_j$ (e eventuali costi di attivazione $CA_j$) per ogni tipologia di cioccolato $j = 1, \ldots, n$. Si assumano $m$ tipologie di vitamine, e sia $R_i$ la quantità di vitamina $i$ richiesta sul singolo afflusso (che ora ammonterà ad una quantità $Q$, non necessariamente il valore di questo parametro $Q$ sarà sempre di 10 Kg). Si indichi con $A_{i,j}$ l'apporto in termini di vitamina $i$ (dove $i = 1, \ldots, m$) per ogni chilo di cioccolato $C_j$ (dove $j = 1, \ldots, n$).

**(8=4+1+1+1+1pt)** Si osservi come il ricorso alla PLI, senza limitarsi alla PL, fosse una rinuncia necessaria dovendo rappresentare i costi di attivazione. Dimostrare questo osservando che il problema coi costi di attivazione è NP-hard, riducendo ad esso 3SAT. La sola descrizione di una riduzione valida vale 4 punti, mentre gli eventuali enunciati del lemma facile e difficile, e le loro eventuali dimostrazioni in breve, valgono 1 punto a testa.

**svolgimento.**

**(3pt)** Quando non ci siano costi di attivazione, introduciamo tre variabili $x_f$, $x_l$, $x_b$ per le quantità di cioccolato fondente, al latte, o bianco, rispettivamente. Si perviene alla seguente formulazione:

$$\min 70\,x_f + 80\,x_l + 60\,x_b$$
$$\begin{cases} x_f + x_l + x_b & = & 10 \\ 400\,x_f + 3000\,x_l + 2200\,x_b & \geq & 11000 \\ 0,6\,x_f + 1\,x_l + 1\,x_b & \geq & 8 \\ 0,6\,x_f + 3\,x_l + 4\,x_b & \geq & 16 \\ 11,4\,x_f + 30\,x_l + 30\,x_b & \geq & 134 \\ 500\,x_f + 700\,x_l + 150\,x_b & \geq & 3000 \\ 24\,x_f + 12\,x_l & \geq & 100 \\ x_f, x_l, x_b \geq 0 \end{cases}$$

**(2pt)** Quando ci siano i costi di attivazione, oltre alle tre variabili $x_f$, $x_l$, $x_b$ per le quantità di cioccolato fondente, al latte, o bianco, da utilizzarsi nel rabbocco, introduciamo inoltre tre variabili booleane $\hat{x}_f$, $\hat{x}_l$, $\hat{x}_b$ intese ad indicare appunto se vi sia impiego del tipo di cioccolato corrispondente. L'idea è che si abbia $\hat{x}_f = 1$ se e solo se $x_f > 0$, e $\hat{x}_f = 0$ altrimenti, quando $x_f = 0$. Ovviamente questo legame andrà forzato conn gli opportuni vincoli; e lo stesso discorso dovrà valere per $x_l$ e $\hat{x}_l$, come anche per $x_b$ e $\hat{x}_b$. La PLI ha una grande capacità espressiva (è NP-hard), un modo semplice per realizzare quanto detto sopra è impiegato nel pervenire alla seguente formulazione:

$$\min 70\,x_f + 80\,x_l + 60\,x_b + 120\,\hat{x}_f + 130\,\hat{x}_l + 101\,\hat{x}_b$$
$$\begin{cases} x_f + x_l + x_b & = & 10 \\ 400\,x_f + 3000\,x_l + 2200\,x_b & \geq & 11000 \\ 0,6\,x_f + 1\,x_l + 1\,x_b & \geq & 8 \\ 0,6\,x_f + 3\,x_l + 4\,x_b & \geq & 16 \\ 11,4\,x_f + 30\,x_l + 30\,x_b & \geq & 134 \\ 500\,x_f + 700\,x_l + 150\,x_b & \geq & 3000 \\ 24\,x_f + 12\,x_l & \geq & 100 \\ x_f \leq 10\,\hat{x}_f \\ x_l \leq 10\,\hat{x}_l \\ x_b \leq 10\,\hat{x}_b \\ x_f, x_l, x_b \geq 0 \\ \hat{x}_f, \hat{x}_l, \hat{x}_b \in \{0,1\} \end{cases}$$

**(1+1=2pt)** Il caso in cui non ci siano costi di attivazione può essere riguardato come il caso particolare di quello con costi di attivazione, come ottenuto quando si abbia $CA_j = 0$ per ogni $j = 1, \ldots, n$. Trattiamo pertanto solo il caso con costi di attivazione.

Introduciamo le $2n$ variabili $x_j, \hat{x}_j$ per $j = 1, \ldots, n$:

$x_j$ è una variabile continua e non-negativa che indica la quantità di cioccolato di tipo $C_j$ da impiegarsi ad ogni rabbocco, espressa in chili;

$\hat{x}_j$ è una variabile booleana, intesa ad agire come variabile indicatore, e caratterizzata dalla proprietà definitoria: $\hat{x}_j = 1$ se e solo se $x_j \neq 0$.

Una volta che si estragga dai dati, la formulazione che proponiamo diviene non solo più generale ma anche più compatta:

$$\min \sum_{j=1}^n CL_j\, x_j + \sum_{j=1}^n CA_j\, \hat{x}_j$$
$$\begin{cases} \sum_{j=1}^n x_j = Q & \\ \sum_{j=1}^n A_{i,j}\, x_j \geq R_j & \text{per ogni } i = 1, \ldots, m \\ x_j \leq Q\, \hat{x}_j & \text{per ogni } j = 1, \ldots, n \\ x_j \geq 0 & \text{per ogni } j = 1, \ldots, n \\ \hat{x}_j \in \{0, 1\} & \text{per ogni } j = 1, \ldots, n \end{cases}$$

La maggiore compattezza della formulazione, una volta espressa con un formalismo più vicino ai costrutti della matematica, ne consente spesso una più chiara lettura ed interpretazione. Anche per questa ragione, oltre che per la maggior generalità e flessibilità, nella pratica si utilizzano linguaggi intermedi di programmazione/modellazione matematica che, come sopra, prevedono l'uso di quantificatori, ma anche costrutti insiemistici e logici, ... e chi più ne ha più ne metta. Si noti inoltre come l'uso di parametri quali $Q$ eviti la presenza di quelli che in programmazione vengono chiamati "numeri magici", l'impiego di $Q$ comporta una taggatura semantica del valore numerico che esso rappresenta, evitando di smarrire il significato di un 10 che, in base al suo solo "face value", potrebbe risultare ambiguo. Al tempo stesso gli $n$ vincoli dove $Q$ va a moltiplicarsi per una variabile indicatore restano comunque dei vincoli lineari in quanto $Q$ è un parametro, non una variabile, il che significa che il suo valore numerico verrà ad esso sostituito in fase di costruzione esplicita del sistema da passare al solver.

**(4pt)** Si assuma data una formula di 3SAT nelle variabili booleane $v_1, \ldots, v_p$ e nelle 3-clausole $T_1, \ldots, T_q$. Una 3-clausola è la congiunzione di 3 letterali sulle variabili $v_1, \ldots, v_p$, come ad esempio $T_t = v_i \vee \overline{v_j} \vee v_k$ che presenta due letterali positivi ed uno (quello sulla variabile $v_j$) negato. Sappiamo che decidere se esista un assegnamento di verità alle $n$ variabili booleane tale da soddisfare ogni clausola (cioè tale che ogni clausola possegga almeno un letterale che valuta a *true*) è un problema NP-completo. Per dimostrare l'NP-hardness del nostro problema di cioccolato (con costi di attivazione) vogliamo rappresentare entro esso la generica istanza di 3SAT data.

Per fare ciò introduciamo $p + q$ vitamine, tutte con richiesta $R_i = 1$, $i = 1, \ldots, p + q$, e le tipologie di cioccolato $C_1, \ldots, C_{2p}$. Prendiamo a zero tutti i costi lineari ed a uno tutti i costi di attivazione, ossia $CL_j = 0$ e $CA_j = 1$ per ogni $j = 1, \ldots, 2p$. Per forzare che almeno uno tra i cioccolati $C_j$ e $C_{j+p}$ trovi impiego, dove $j = 1, \ldots, p$, poniamo $A_{j,x} = 0$ per ogni $x = 1, \ldots, 2p$, tranne che $A_{j,j} = A_{j,j+p} = 1$. (La richiesta sulla vitamina $j$ forza così che $\hat{x}_j + \hat{x}_{j+p} \geq 1$). Già a questo punto il valore della funzione obiettivo dovrà essere almeno $p$ (poichè $\hat{x}_j + \hat{x}_{j+p} \geq 1$ per ogni $j = 1, \ldots, p$), e l'ottimo potrà raggiungere questo valore minimo solo su eventuali soluzioni con $\hat{x}_j + \hat{x}_{j+p} = 1$ per ogni $j = 1, \ldots, p$. Quando ci chiediamo se sia possibile ottenere tale valore minimo, stiamo quindi forzando una scelta binaria tra l'impiego del cioccolato $C_j$ e l'impiego del cioccolato $C_{j+p}$ per ogni $j = 1, \ldots, p$. Abbiamo quindi già i nostri "componenti bistabili" atti a rappresentare scelte binarie, ci manca solo rappresentare le clausole, ed a questo scopo impiegheremo le $q$ successive vitamine, dalla $p+1$ alla $p+q$. Per rappresentare la 3-clausola $T_t$, $t = 1, \dots, q$, poniamo $A_{p+t,x} = 0$ per ogni indice $x = 1, \dots, 2p$, tranne quei 3 indici che corrispondano ai letterali presenti nella clausola come segue:

se il letterale positivo $v_i$ compare nella 3-clausola $T_t$, allora $A_{p+t,i} = 1$; se il letterale negato $\overline{v_i}$ compare nella 3-clausola $T_t$, allora $A_{p+t,p+i} = 1$.

Per il parametro $Q$ fissiamo il valore $Q := p$, anche se in realtà andrebbe bene qualsiasi valore non inferiore a $p$. (Domanda: perchè con $Q < p$ la nostra riduzione non funzionerebbe?) La descrizione della riduzione è ora completa, e merita già l'attribuzione dei 4 punti in palio.

Lo statement dei due lemmi cardine che sostengono la riduzione assicurano un ulteriore punto ciascuno.

**(1pt)** Lemma facile: se la 3SAT formula di partenza è soddisfacibile allora esiste una soluzione di costo $p$ per l'istanza cioccolatosa costruita come descritto.

**(1pt)** Lemma difficile: il minimo costo per l'istanza cioccolatosa costruita è almeno $p$. Inoltre, se esso è precisamente $p$ allora la 3SAT formula di partenza è soddisfacibile.

Anche se in generale non vi sarà il modo più conveniente di raccogliere punti, esemplifico delle possibili brevi dimostrazioni dei due lemmi.

**($\geq$1pt)** *Proof del Lemma facile:* Sia $\Phi : \{v_1, \dots, v_p\} \mapsto \{0, 1\}$ un assegnamento di verità alle variabili tale da soddisfare ogni clausola della forma di 3SAT data. Mostreremo come derivare da $\Phi$ una soluzione ammissibile di costo $p$ per l'istanza cioccolatosa costruita. Per ogni $j = 1, \dots, p$, si effettuino le seguenti scelte:

**se** $\Phi(v_j) = 1$ si prenda $x_j = \hat{x}_j = 1$ e $x_{p+j} = \hat{x}_{p+j} = 0$;

**se** $\Phi(v_j) = 0$ si prenda $x_j = \hat{x}_j = 0$ e $x_{p+j} = \hat{x}_{p+j} = 1$.

È facile verificare che il valore della soluzione descritta dalle $x_j$ e dalle $\hat{x}_j$, come da nostra formulazione PLI per il problema cioccolatoso proposta ad un punto precedente del presente esercizio, è precisamente $p$; rimane pertanto solo da verificare che tale soluzione è ammissibile, ossia soddisfa tutti i vincoli della nostra formulazione di PLI. In primo luogo, tutte le variabili sono non-negative, e di fatto a valori 0/1. Disegue poi immediatamente da come la soluzione è stata definita che $x_j + x_{p+j} = 1$ per ogni $j = 1, \dots, p$, quindi i vincoli di richiesta sulle prime $p$ vitamine sono soddisfatti, come anche il primo vincolo $\sum_{j=1}^{n} x_j = \sum_{j=1}^{2p} x_j = Q$ della nostra formulazione di PLI. Inoltre vi è la coerenza richiesta tra i valori di $x_j$ e della sua variabile indicatore $\hat{x}_j$, ossia sono soddisfatti anche i vincoli $x_j \leq Q\,\hat{x}_j$, dove $Q = p \geq 1$. Infine, per ogni $j = p+1, \dots, p+q$, il vincolo di richiesta sulla vitamina $j$ è soddisfatto in quanto la 3-clausola $T_{j-p}$ possiede almeno un letterale che valuta a *true* sotto $\Phi$ avendo noi assunto che $\Phi$ soddisfi ogni clausola. (Di fatto il vincolo di richiesta sulla vitamina $j$ è stato introdotto proprio allo scopo di rappresentare la 3-clausola $T_{j-p}$.) $\square$

**($\geq$1pt)** *Proof del Lemma difficile:* Si consideri una qualsiasi soluzione ammissibile per l'istanza cioccolatosa da noi costruita partendo dall'istanza di 3SAT. Si assuma che tale soluzione sia descritta con la specifica dei valori delle variabili $x_j$ e dalle variabili $\hat{x}_j$, come da nostra formulazione PLI del problema cioccolatoso. Se la soluzione è ammissibile, essa deve allora soddisfare i vincoli di richiesta per le prime $p$ vitamine, e quindi, per ogni $j = 1, \dots, p$ si avrà che $\hat{x}_j + \hat{x}_{p+j} \geq 1$ poichè $x_j + x_{p+j} \geq 1$. Da questo, e considerata la funzione obiettivo, disegue immediatamente la prima asserzione nello statement del lemma. Inoltre, se il valore della soluzione ammissibile considerata è precisamente $p$, allora necessariamente $\hat{x}_j + \hat{x}_{p+j} = 1$ per ogni $j = 1, \dots, p$. Si consideri l'assegnamento di verità $\Phi : \{v_1, \dots, v_p\} \mapsto \{0, 1\}$ con $\Phi(v_j) = true$ se e solo se $\hat{x}_j = 1$. Si noti che vale il seguente:

**se** $\Phi(v_j) = 1$ allora $\hat{x}_j = 1$ e $x_{p+j} = \hat{x}_{p+j} = 0$;

**se** $\Phi(v_j) = 0$ allora $x_j = \hat{x}_j = 0$ e $x_{p+j} = \hat{x}_{p+j} = 1$.

Pertanto, per ogni $j = p+1, \ldots, p+q$, dato che il vincolo di richiesta sulla vitamina $j$ è soddisfatto, allora la 3-clausola $T_{j-p}$ possiede almeno un letterale che valuta a *true* sotto $\Phi$. $\square$

---

**Problema 2 (5 punti):**
Stabilire la veridicità delle seguenti affermazioni argomentando compiutamente le tue risposte (fare riferimento ad enunciati noti ove positive, proporre controesempi ove negative).

*Se un problema di PL in forma standard ha una soluzione ottima non di base allora tale problema*

*2.1.* **(1pt)** *È necessariamente degenere ossia ha almeno una soluzione di base degenere.*

*2.2.* **(1pt)** *Ha un duale che è necessariamente ammissibile.*

*2.3.* **(1pt)** *Ha un duale che è necessariamente limitato.*

*2.4.* **(1pt)** *Ha almeno una soluzione di base ottima.*

*2.5.* **(1pt)** *Ha almeno due soluzioni di base ottime.*

**risposte.**

2.1. È necessariamente degenere ossia ha almeno una soluzione di base degenere. FALSO - si consideri il problema $\max\{x_1 : 0 \leq x_1 \leq 1, 0 \leq x_2 \leq 1\}$.

2.2. Ha un duale che è necessariamente ammissibile. VERO - per il teorema della dualità forte. Se il duale fosse inammissibile il primale potrebbe essere solamente inammissibile od illimitato.

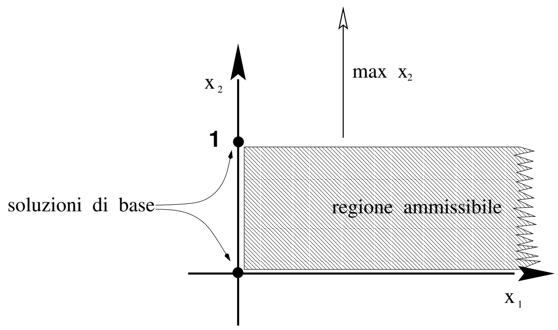
2.3. Ha un duale che è necessariamente limitato. VERO - basta il teorema della dualità debole.

2.4. Ha almeno una soluzione di base ottima. VERO - segue dal teorema fondamentale della PL. In pratica, è una conseguenza dell'algoritmo del simplesso.

2.5. Ha almeno due soluzioni di base ottime. FALSO - un controesempio è dato dal seguente problema in forma standard.

$$\max \ x_2$$
$$\begin{cases} x_2 \leq 1 \\ x_1, x_2 \geq 0 \end{cases}$$

Esso presenta un'unica soluzione di base ottima nonostante ammetta soluzioni ottime non di base.




---

**Problema 3 (7 punti):**

Un robot $R$, inizialmente situato nella cella A–1, deve portarsi nella sua home $H$ situata nella cella G–9.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| $A$ | $R$ | 1 | 3 | 1 | 1 | 1 | 0 | 0 | • |
| $B$ | 2 | 2 | 1 | 0 | • | • | 0 | 0 | 0 |
| $C$ | 2 | 2 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| $D$ | 0 | 0 | • | 0 | 0 | 0 | 1 | • | 0 |
| $E$ | 0 | 0 | 1 | 1 | • | 1 | 0 | 0 | 0 |
| $F$ | 0 | 1 | 1 | 1 | 0 | 3 | • | 0 | 1 |
| $G$ | 3 | • | 0 | 1 | 2 | 0 | 0 | 1 | $H$ |

I movimenti base possibili sono il passo verso destra (ad esempio dalla cella A–3 alla cella A–4) ed il passo verso in basso (ad esempio dalla cella A–3 alla cella B–3). Tuttavia il robot non può visitare le celle occupate da un pacman (•). Quanti sono i percorsi possibili? Inoltre, in ogni cella non occupata da un pacman (•) é presente un valore intero che esprime un pedaggio che viene pagato dal robot se passa per quella cella. Potremmo quindi essere interessati al minimizzare il costo complessivo della traversata.

**3.1(1pt)** Quanti sono i percorsi possibili se la partenza è in A–1?

**3.2 (1pt)** e se la partenza è in B–3?

**3.3 (1pt)** e se con partenza in A–1 il robot deve giungere in F–6?

**3.4 (1pt)** e se con partenza in A–1 ed arrivo in G–9 al robot viene richiesto di passare per la cella D–5?

**3.5(1pt)** Quale é il minimo costo di una traversata da A–1 a G–9?

**3.6(2pt)** Quanti sono i percorsi possibili che comportano questo costo minimo?

**svolgimento.** La risposta alle prime due domande può essere reperita nella rispettiva cella della seguente tabella di programmazione dinamica, dove in ogni cella C, partendo da quelle in basso a destra, si é computato il numero di percorsi che vanno dalla cella C alla cella G–9.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| $A$ | 347 | 187 | 92 | 36 | 8 | 8 | 8 | 2 | • |
| $B$ | 160 | 95 | 56 | 28 | • | • | 6 | 2 | 1 |
| $C$ | 65 | 39 | 28 | 28 | 18 | 11 | 4 | 1 | 1 |
| $D$ | 26 | 11 | • | 10 | 7 | 7 | 3 | • | 1 |
| $E$ | 15 | 11 | 7 | 3 | • | 4 | 3 | 3 | 1 |
| $F$ | 4 | 4 | 4 | 3 | 2 | 1 | • | 2 | 1 |
| $G$ | 0 | • | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Per rispondere alle due seguenti domande compilo un'ulteriore tabella, dove in ogni cella C, partendo da quelle in alto a sinistra, si computa il numero di percorsi che vanno dalla cella A–1 alla cella C.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| $A$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | • |
| $B$ | 1 | 2 | 3 | 4 | • | • | 1 | 2 | 2 |
| $C$ | 1 | 3 | 6 | 10 | 10 | 10 | 11 | 13 | 15 |
| $D$ | 1 | 4 | • | 10 | 20 | 30 | 41 | • | 15 |
| $E$ | 1 | 5 | 5 | 15 | • | 30 | 71 | 71 | 86 |
| $F$ | 1 | 6 | 11 | 26 | 26 | 56 | • | 71 | 157 |
| $G$ | 1 | • | 11 | 37 | 63 | 119 | 119 | 190 | 347 |

Ritrovare il valore 347 ci conforta. La risposta alla terza domanda è contenuta nella rispettiva cella di questa seconda tabella.
La quarta domanda richiede di combinare le informazioni provenienti dalle due tabelle: la risposta è ottenuta come prodotto dei due valori riportati nelle due tabelle entro la cella di passaggio obbligato per il robot.

Per rispondere alle ultime due domande compilo un'ulteriore tabella, dove in ogni cella C, partendo da quelle in alto a sinistra, si computa il minimo costo di un percorso che va dalla cella A–1 alla cella C. Computiamo e riportiamo inoltre in piccolo, per ogni cella C, il numero di tali percorsi di costo minimo.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| $A$ | $0_1$ | $1_1$ | $4_1$ | $5_1$ | $6_1$ | $7_1$ | $7_1$ | $7_1$ | • |
| $B$ | $2_1$ | $3_1$ | $4_1$ | $4_1$ | • | • | $7_1$ | $7_2$ | $7_2$ |
| $C$ | $4_1$ | $5_1$ | $4_1$ | $5_2$ | $5_2$ | $5_2$ | $6_2$ | $7_2$ | $8_4$ |
| $D$ | $4_1$ | $4_1$ | • | $5_2$ | $5_4$ | $5_6$ | $6_6$ | • | $8_4$ |
| $E$ | $4_1$ | $4_2$ | $5_2$ | $6_4$ | • | $6_6$ | $6_{12}$ | $6_{12}$ | $6_{12}$ |
| $F$ | $4_1$ | $5_3$ | $6_5$ | $7_9$ | $7_9$ | $9_6$ | • | $6_{12}$ | $7_{24}$ |
| $G$ | $7_1$ | • | $6_5$ | $7_5$ | $9_{14}$ | $9_{20}$ | $9_{20}$ | $7_{12}$ | $7_{36}$ |

Leggendo i valori riportati nella cella G–9 scopriamo che il minimo costo di una traversata é di 7, e che esistono 36 diversi possibili percorsi per raccogliere questo valore.

Riportiamo quindi i risultati finali.

| consegna | numero percorsi |
|---|---|
| A–1 $\to$ G–9 | 347 |
| B–3 $\to$ G–9 | 56 |
| A–1 $\to$ F–6 | 56 |
| passaggio per D–5 | 140 |
| minimo costo | 7 |
| numero di min-cost paths | 36 |

Per maggiori e precise informazioni sulla logica con cui siano state compilate le varie tabelle di programmazione dinamica rimandiamo al codice c++ che le ha prodotte. Esso è reso disponibile nella stessa cartella della presente correzione.

---

**Problema 4 (1+6·1=7 punti):**

Una tabella di programmazione dinamica è stata compilata per il problema di knapsack con zaino di capacità $B = 35$ e con riferimento al set di oggetti nella legenda in calce alla tabella stessa.

## Tabella di Programmazione Dinamica per il problema dello Zaino

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| @(0,0) | 0 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | |
| $A(16,52)$ | 0 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | 52 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × |
| $B(15,51)$ | 0 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | 51 | 52 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | 103 | | | | |
| $C(13,40)$ | 0 | × | × | × | × | × | × | × | × | × | × | × | × | 40 | × | 51 | 52 | × | × | × | × | × | × | × | × | × | × | × | | | | | | | | × |
| $D(11,31)$ | 0 | × | × | × | × | × | × | × | × | × | × | 31 | × | 40 | × | 51 | 52 | × | × | × | × | × | × | × | 71 | × | 82 | 83 | 91 | 92 | × | 103 | × | × | × | × |
| $E(8,20)$ | 0 | × | × | × | × | × | × | × | 20 | × | × | 31 | × | 40 | × | 51 | 52 | × | × | 51 | × | 60 | × | 71 | 72 | × | | | | | × | | 91 | × | 102 | 103 |
| $F(7,21)$ | 0 | × | × | × | × | × | × | 21 | 20 | × | × | 31 | × | 40 | × | 51 | 52 | × | 52 | 51 | 61 | 60 | 72 | 73 | 72 | × | | | | 92 | 92 | 103 | 91 | 103 | 104 | 112 |
| $G(5,16)$ | 0 | × | × | × | × | 16 | × | 21 | 20 | × | × | 31 | 37 | 40 | × | 51 | 52 | × | 56 | 51 | 67 | 68 | 72 | 73 | 72 | 77 | 82 | 88 | 91 | | 92 | 103 | | | | |
| $H(3,5)$ | 0 | × | × | 5 | × | 16 | × | 21 | 21 | × | 26 | 31 | 37 | 40 | 36 | 51 | 52 | × | 56 | 57 | 67 | 68 | 72 | 73 | 73 | 77 | 82 | 88 | 91 | 92 | 93 | | | | | |
| $I(2,15)$ | 0 | × | 15 | 5 | × | 20 | × | 31 | 21 | 36 | 36 | 31 | 41 | 46 | 52 | 55 | 52 | 66 | 67 | 57 | 71 | 72 | 82 | 83 | 87 | 88 | 88 | 92 | 97 | 103 | 106 | 107 | | | | |
| $J(1,7)$ | 0 | 7 | 15 | 22 | 12 | 20 | 27 | 31 | 38 | 36 | 43 | 43 | 41 | 48 | 53 | 59 | 62 | 66 | 73 | 74 | 71 | 78 | 82 | 89 | 90 | 94 | 95 | 95 | 99 | 104 | | | | | | |

(come stilata in riferimento ai seguenti oggetti)

| nome | A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|---|
| peso | 16 | 15 | 13 | 11 | 8 | 7 | 5 | 3 | 2 | 1 |
| valore | 52 | 51 | 40 | 31 | 20 | 21 | 16 | 5 | 15 | 7 |

## Tabella delle Risposte

| B | vincolo agg. | max val | peso | quali prendere |
|---|---|---|---|---|
| 35 | - | | | |
| 35 | evita J | | | |
| 35 | prendi J, evita I | | | |
| 35 | evita I e J | | | |
| 33 | prendi J | | | |
| 32 | evita I e J | | | |

Purtroppo, alcune entry della tabella si sono sbiadite. Si colmino TUTTE (**1pt**) le lacune nella tabella di programmazione dinamica e si identifichino correttamente le risposte (**6pt**) riportandole nella tabella delle risposte.

**svolgimento.**

Qui occorre innanzitutto completare diligentemente la tabella di programmazione dinamica. Una buona parte delle entry sono facili da ricostruire in sicurezza avvalendosi di proprietà di monotonia od altre considerazioni. Sulle altre converrà muoversi con piedi di piombo per non inficiare anche tutte le risposte a seguire.

(Ovviamente, se vi aiuta, potete farvi vostri conteggi o prove nella brutta, o lavorare in matita, prima di sporcare il foglio con le tabelle messovi a disposizione).

Per maggiori e precise informazioni sulla logica con cui vada compilata la tabella di programmazione dinamica rimandiamo al codice c++ che la ha prodotta. Esso è reso disponibile nella stessa cartella della presente correzione.

# Tabella di Programmazione Dinamica per il problema dello Zaino

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $@(0,0)$ | 0 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × |
| $A(16,52)$ | 0 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | 52 | × |
| $B(15,51)$ | 0 | × | × | × | × | × | × | × | × | × | × | × | × | × | × | 51 | 52 | × |
| $C(13,40)$ | 0 | × | × | × | × | × | × | × | × | × | × | × | × | 40 | × | 51 | 52 | × |
| $D(11,31)$ | 0 | × | × | × | × | × | × | × | × | × | × | 31 | × | 40 | × | 51 | 52 | × |
| $E(8,20)$ | 0 | × | × | × | × | × | × | × | 20 | × | × | 31 | × | 40 | × | 51 | 52 | × |
| $F(7,21)$ | 0 | × | × | × | × | × | × | 21 | 20 | × | × | 31 | × | 40 | × | 51 | 52 | × |
| $G(5,16)$ | 0 | × | × | × | × | 16 | × | 21 | 20 | × | × | 31 | 37 | 40 | × | 51 | 52 | × |
| $H(3,5)$ | 0 | × | × | 5 | × | 16 | × | 21 | 21 | × | 26 | 31 | 37 | 40 | 36 | 51 | 52 | × |
| $I(2,15)$ | 0 | × | 15 | 5 | × | 20 | × | 31 | 21 | 36 | 36 | 31 | 41 | 46 | 52 | 55 | 52 | 66 |
| $J(1,7)$ | 0 | 7 | 15 | 22 | 12 | 20 | 27 | 31 | 38 | 36 | 43 | 43 | 41 | 48 | 53 | 59 | 62 | 66 |

| | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $@(0,0)$ | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × |
| $A(16,52)$ | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × | × |
| $B(15,51)$ | × | × | × | × | × | × | × | × | × | × | × | × | × | 103 | × | × | × | × |
| $C(13,40)$ | × | × | × | × | × | × | × | × | × | × | 91 | 92 | × | 103 | × | × | × | × |
| $D(11,31)$ | × | × | × | × | × | × | 71 | × | 82 | 83 | 91 | 92 | × | 103 | × | × | × | × |
| $E(8,20)$ | × | 51 | × | 60 | × | 71 | 72 | × | 82 | 83 | 91 | 92 | × | 103 | 91 | × | 102 | 103 |
| $F(7,21)$ | 52 | 51 | 61 | 60 | 72 | 73 | 72 | × | 82 | 83 | 91 | 92 | 92 | 103 | 91 | 103 | 104 | 112 |
| $G(5,16)$ | 56 | 51 | 67 | 68 | 72 | 73 | 72 | 77 | 82 | 88 | 91 | 92 | 92 | 103 | 99 | 107 | 108 | 112 |
| $H(3,5)$ | 56 | 57 | 67 | 68 | 72 | 73 | 73 | 77 | 82 | 88 | 91 | 92 | 93 | 103 | 99 | 107 | 108 | 112 |
| $I(2,15)$ | 67 | 57 | 71 | 72 | 82 | 83 | 87 | 88 | 88 | 92 | 97 | 103 | 106 | 107 | 108 | 118 | 114 | 122 |
| $J(1,7)$ | 73 | 74 | 71 | 78 | 82 | 89 | 90 | 94 | 95 | 95 | 99 | 104 | 110 | 113 | 114 | 118 | 125 | 122 |

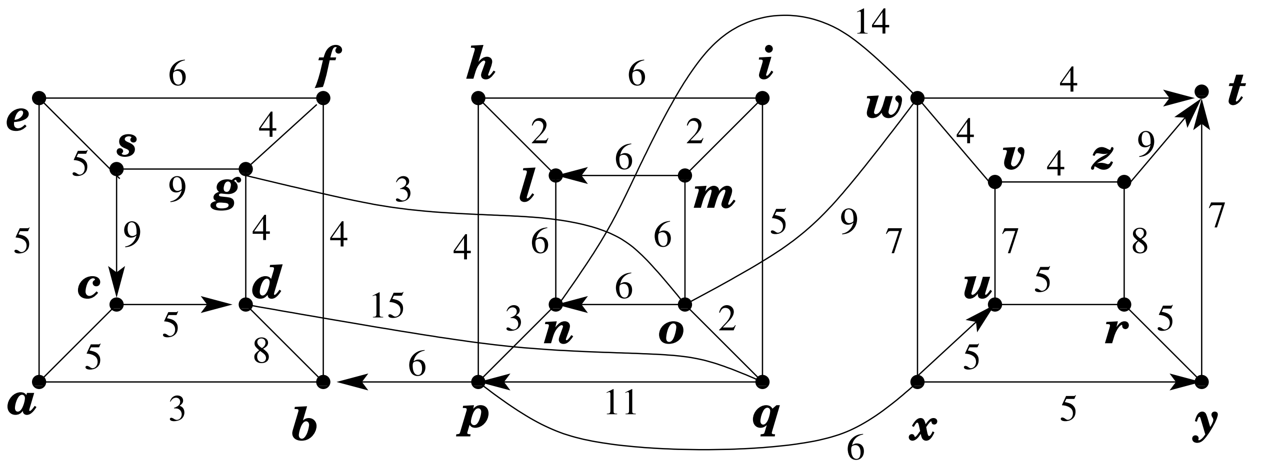
(come stilata in riferimento ai seguenti oggetti)

| nome | A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|---|
| peso | 16 | 15 | 13 | 11 | 8 | 7 | 5 | 3 | 2 | 1 |
| valore | 52 | 51 | 40 | 31 | 20 | 21 | 16 | 5 | 15 | 7 |

# Tabella delle Risposte

| B | vincolo agg. | max val | peso | quali prendere |
|---|---|---|---|---|
| 35 | - | 125 = 52+51+15+7 | 34 = 16+15+2+1 | A,B,I,J |
| 35 | evita J | 122 = 51+40+16+15 | 35 = 15+13+5+2 | B,C,G,I |
| 35 | prendi J, evita I | 115 = 52+40+16+7 | 35= 16+13+5+1 | A,C,G,J |
| 35 | evita I e J | 112 = 51+40+21 | 35 = 15+13+7 | B,C,F |
| 33 | prendi J | 115 = 51+21+16+5+15+7 | 33 = 15+7+5+3+2+1 | B,F,G,H,I,J |
| 32 | evita I e J | 103 = 52+51 | 31 = 16+15 | A,B |

**Problema 5 (23 punti):**
Si consideri il grafo $G$, con pesi sugli archi, riportato in figura.




5.1.**(2pt)** Dire, certificandolo, (1) se il grafo $G$ è planare oppure no **(1pt)**; (2) se il grafo $G'$ ottenuto da $G$ rimpiazzando l'arco $go$ con l'arco $gm$ è planare oppure no **(1pt)**.

5.2.**(4pt)** Fornendo i certificati del caso, dire quale sia il minimo numero di archi la cui rimozione renda bipartito: il grafo $G$ **(1pt)**; il grafo $G'$ **(1pt)**. Argomentare l'unicità del minimo insieme di archi da rimuovere nel caso di $G$ **(1pt)** e di $G'$ **(1pt)**.

5.3.**(1pt)** Trovare un albero ricoprente di $G$ di peso minimo.

5.4.**(4pt)** (**(1pt)**) Di quanto è necessario diminuire il peso dell'arco $wo$ affinchè esso appartenga a qualche soluzione ottima (ma non tutte, fornire quindi ambo i certificati (**(1pt)**))? (**(1pt)**) Di quanto è necessario aumentare il peso dell'arco $wv$ affinchè esso smetta di appartenere a tutte le soluzioni ottime (a quel punto apparterà ad alcune ma non tutte, puoi esibire due certificati per sigillare la tua risposta (**(1pt)**))?

5.5.**(1pt)** Trovare tutti gli alberi ricoprenti di peso minimo. (Dire quanti sono e specificare con precisione come generarli).

5.6.**(1pt)** Trovare un albero dei cammini minimi da $s$ e determinare le distanze di tutti i nodi da $s$.

5.7.**(1pt)** Trovare tutti gli alberi dei cammini minimi da $s$. (Dire quanti sono e specificare con precisione come generarli).

5.8.**(2pt)** Trovare un massimo flusso dal nodo $s$ al nodo $t$.

5.9.**(1pt)** Certificare l'ottimalità del flusso massimo dal nodo $s$ al nodo $t$.

5.10.**(1pt)** Quanti sono i possibili tagli minimi?

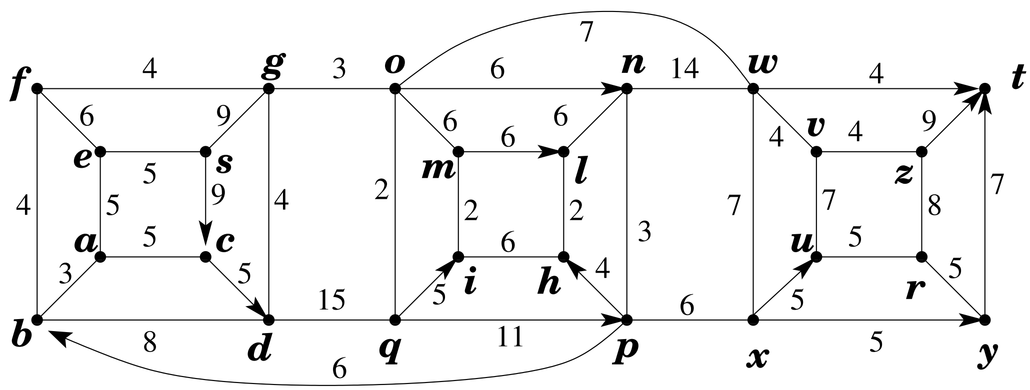
5.11.**(1pt)** Quale è il minimo numero di archi su cui violare il vincolo di capacità per riuscire ad incrementare il flusso massimo? Quali sono questi archi?

5.12.**(1+1pt)** Argomentare che la scelta al punto precedente è ottima. Argomentare che è unica.

5.13.**(1+1pt)** Flusso massimo quando si sia rimosso il vincolo di capacità su questi archi e suo certificato di ottimalità.

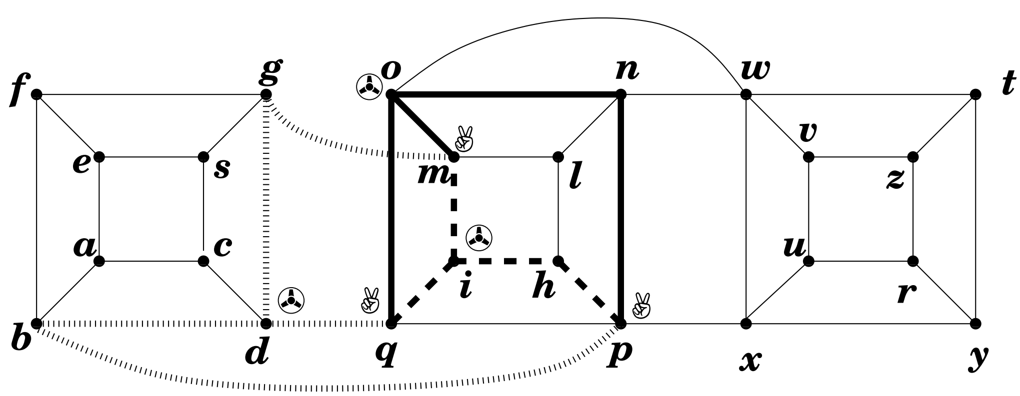
**risposte.**

Il fatto che $G$ sia planare può essere messo in evidenza esibendo il planar embedding in figura.

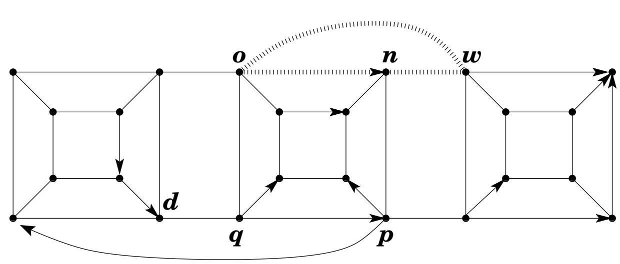



Nello svolgimento dei successivi punti converrà riferirsi al planar drawing fornito sopra.

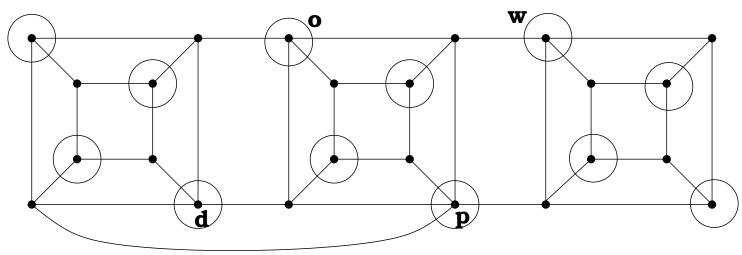
Il fatto che $G'$ **non** sia planare può essere messo in evidenza esibendone un sottografo che sia isomorfo ad una $K_{3,3}$ o $K_5$ subdivision. Nella seguente figura esibiamo una $K_{3,3}$ subdivision (è facile convincersi che $G'$ non possa contenere alcuna $K_5$ subdivision).



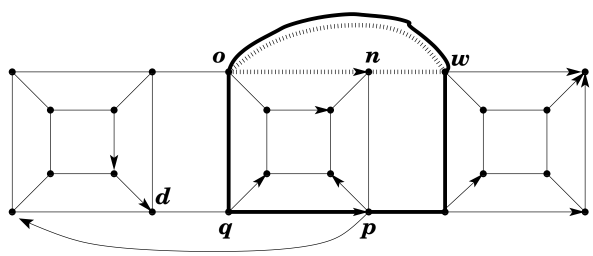

Il fatto che $G$ non sia bipartito, e che sia quindi richiesta la rimozione di almeno un arco per renderlo tale, è certificato dal ciclo dispari (triangolo $onw$) rappresentato in figura.



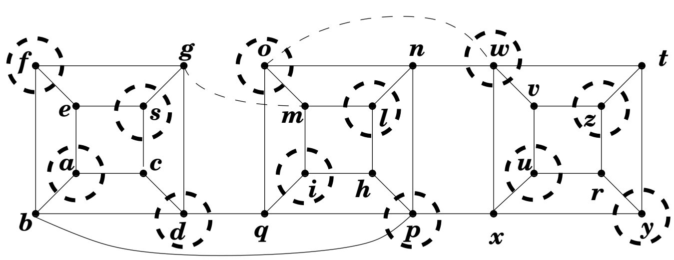
In effetti la rimozione del solo arco ($ow$) basta a rendere $G$ bipartito come esibito nella seguente figura.



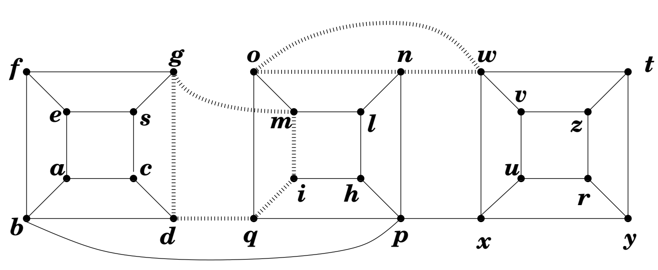

Di fatto $ow$ è l'unico arco la cui rimozione rende il grafo bipartito, data l'esistenza di due cicli dispari con il solo arco $ow$ in comune:



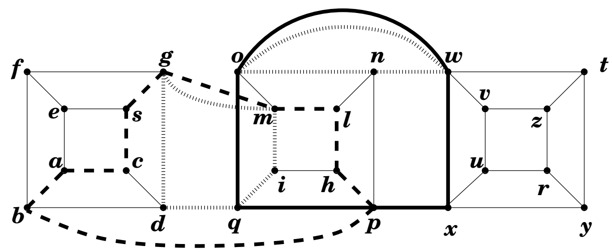

Il grafo $G'$ ottenuto da $G$ rimpiazzando l'arco $go$ con l'arco $gm$ può essere reso bipartito rimuovendo due archi come indicato nella seguente figura (la figura contiene anche la bicolorazione valida, atta a certificare che $G'$ sia bipartito a valle della rimozione dei due archi).



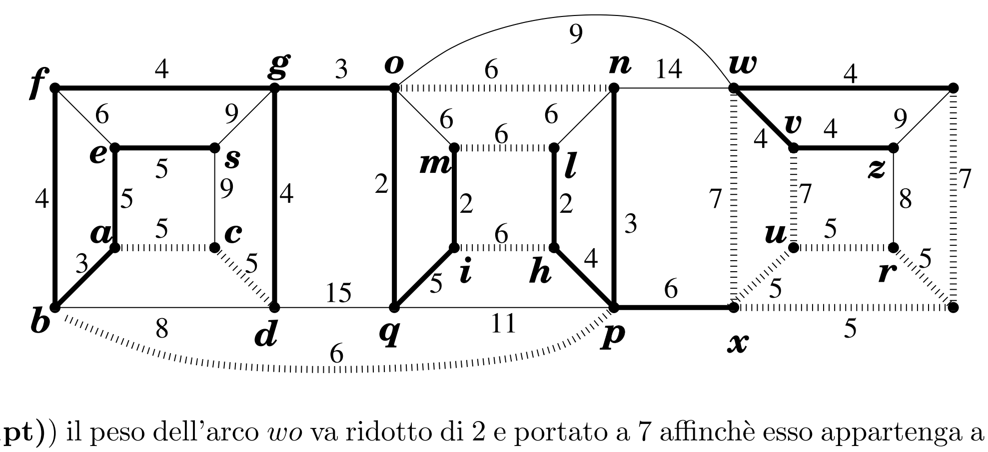

La seguente figura esibisce due cicli dispari di $G'$, disgiunti sugli archi. Essi costituiscono prova o evidenza (certificati) che almeno due archi dovevano essere rimossi al fine di rendere $G'$ bipartito.



Anche in questo caso rimuovere gli archi $ow$ e $gm$ rappresenta l'unico modo di rendere il grafo bipartito. Questo può essere argomentato esibendo i quattro cicli dispari nella seguente figura:




La seguente figura esprime la famiglia degli alberi ricoprenti di peso minimo. Ci sono $2 \cdot 4 \cdot 3 \cdot 4 = 96$ alberi ricoprenti di perso minimo e ciascuno di essi include i 17 archi in linea spessa, più uno qualsiasi dei 2 archi di peso 5 incidenti al nodo $c$ (i 2 archi in linea sfumata spessa presenti nella zona a sinistra), più uno qualsiasi dei 4 archi di peso 6 in linea sfumata spessa presenti nella zona centrale (gli archi $on$, $ml$, $ih$, $pd$), più uno qualsiasi dei 3 archi di peso 7 in linea sfumata spessa presenti nella zona a destra (infatti, se nel grafo $G$ contraiamo tutti gli archi di peso inferiore a 7 e rimuoviamo tutti gli archi di peso superiore a 7 ci ritroviamo con 2 soli nodi connessi da questi 4 archi disposti in parallelo), più **3** qualsiasi dei 4 archi di peso 5 in linea sfumata spessa presenti nella zona a destra (infatti, se nel grafo $G$ contraiamo tutti gli archi di peso inferiore a 5 e rimuoviamo tutti gli archi di peso superiore a 5 ci ritroviamo con una componente connessa che è un quadrato di questi 4 archi. (La componente connessa di 2 nodi connessi da 2 archi paralleli evidenzia l'intercambiabilità dei 2 archi di peso 5 incidenti al nodo $c$ di cui si era detto più sopra).



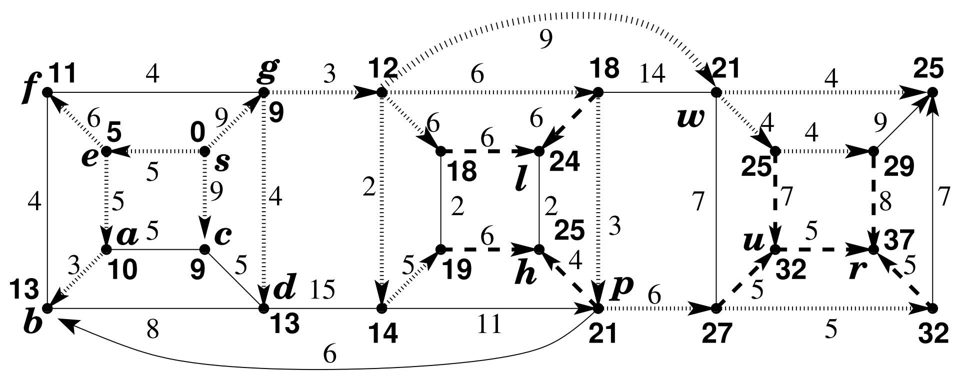

(**(1pt)**) il peso dell'arco $wo$ va ridotto di 2 e portato a 7 affinchè esso appartenga a qualche soluzione ottima.

(**(1pt)**) Quando il peso di $wo$ fosse 7, esso sarebbe un arco di peso minimo nel taglio che separa dal resto del grafo i 4 nodi $w$,$v$,$z$,$t$ (quelli in alto nel cubo di destra), e pertanto apparterrebbe sicuramente a qualche soluzione ottima in virtù del lemma del taglio. La riduzione non può essere inferiore a 2 poichè anche quando $wo$ pesasse 7 esisterebbero soluzioni ottime che non lo utilizzano, come certificato dal ciclo $wxphimo$, dove $wo$ è arco di peso massimo, e in virtù del lemma del ciclo.

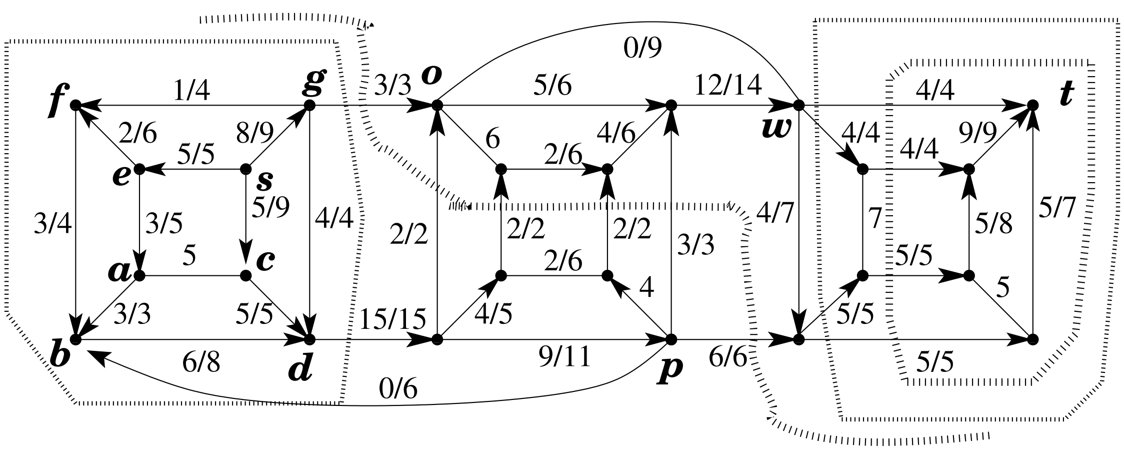
(**(1pt)**) il peso dell'arco $wv$ va aumentato di 3 e portato a 7 affinchè esso smetta di appartenere a tutte le soluzioni ottime.

(**(1pt)**) Quando il peso di $wv$ fosse 7, esso sarebbe un arco di peso minimo nel ciclo $vuxw$ e pertanto esisterebbero sicuramente soluzioni ottime che lo evitano in virtù del lemma del ciclo. La riduzione non può essere inferiore a 3 poichè anche quando $wv$ pesasse 7 esisterebbero soluzioni ottime che lo utilizzano, come certificato dal taglio che separa dal resto del grafo i 2 nodi $v$,$z$, dove $wv$ è arco di peso minimo, e in virtù del lemma del taglio.

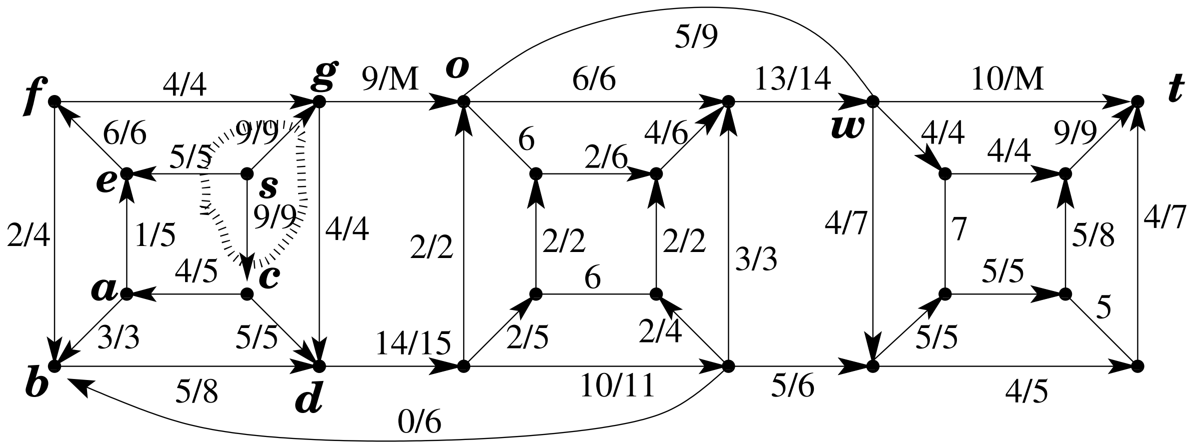
La seguente figura esprime la famiglia degli alberi dei cammini minimi dal nodo $s$. Ci sono $2^3 \cdot 3 = 24$ alberi dei cammini minimi dal nodo $s$ e ciascuno di essi include i 19 archi in linea spessa, più uno qualsiasi dei 2 archi tratteggiati entranti nel nodo $h$, uno qualsiasi dei 2 archi tratteggiati entranti nel nodo $l$, e uno qualsiasi dei 2 archi tratteggiati entranti nel nodo $u$, e uno qualsiasi dei 2 archi tratteggiati entranti nel nodo $r$ (quello a massima distanza, ossia 37, dal nodo $s$).




La seguente figura esibisce un flusso massimo (non esibisco tutti i passaggi che ho dovuto compiere per ottenerlo) ed un taglio (minimo) che ne dimostra l'ottimalità.




Il flusso ha valore 18 e satura diversi archi del grafo. La rete ausiliaria, ed il problema di raggiungibilità ad essa associato, consentono forse di individuare più chiaramente i 4 $s,t$-tagli minimi (tutti gli archi uscenti sono saturati e solo gli archi sgomberi possono entrare, pertanto il valore del flusso netto per essi coincide con la capacità del taglio uscente) che abbiamo riportato nella stessa figura. Alcuni di questi tagli sono disgiunti sugli archi uscenti; pertanto, se vogliamo aumentare il valore del flusso, dobbiamo entrare nell'ordine di idee di violare i vincoli di capacità per almeno due archi. Si noti che tutti e 4 questi tagli (certificati di ottimalità del flusso) contengono almeno uno tra gli archi $go$ e $wt$, questi sono pertanto i due archi su cui decido di forare le capacità per ottenere un incremento del valore del flusso massimo. Il flusso massimo diviene a questo punto quello illustrato nella seguente figura, insieme al nuovo certificato di ottimalità.




Tuttavia, contrariamente a quanto suggerito dalla domanda nel testo, questa scelta di due archi non è unica in quanto avrei potuto altresì scegliere *go* e *xy*. Il fatto che queste siano le due sole scelte possibili disegue da: (1), esistono due di questi 4 tagli che tra di loro condividono solo l'arco *go* e non hanno alcun arco in comune con gli altri due tagli (questo già forza la scelta sull'arco *go*); e (2), gli altri due tagli condividono solo gli archi *wt* e *xy*.