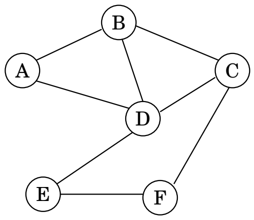
Nome: .... Cognome: ....

Matricola: .... Firma: ....

Esame di Ricerca Operativa - 9 febbraio 2010
Facoltà di Architettura - Udine

Problema 1 (2+2 punti): Un Insieme Indipendente in un grafo $G = (V, E)$ è un sottoinsieme $X \subseteq V$ tale che per ogni due nodi $x_1, x_2 \in X$ non vi è in G alcun arco $x_1 x_2$. Quando ad ogni nodo v è associato un valore w_v, allora il valore di $X \subseteq V$ è espresso da $val(X) := \sum_{v \in X} w_v$.

Ad esempio, con riferimento al grafo G in figura, gli insiemi $\{A, C, E\}$ e $\{B, F\}$ sono due possibili Insiemi Indipendenti entrambi di valore 7.



	A	B	C	D	E	F
Valore	2	3	2	4	3	4

Nelle applicazioni siamo solitamente interessati a trovare Insiemi Indipendenti dwl massimo valore possibile.

Formulare come un problema di Programmazione Lineare Intera (PLI) la ricerca di un Insieme Indipendente di valore massimo nel grafo G in figura.

Mostrare come sia più in generale possibile formulare come un problema di Programmazione Lineare Intera (PLI) la ricerca di un massimo Insieme Indipendente di massimo valore su un grafo $G = (V, E)$ generico.

Problema 2 (4 punti):

Una pasticceria produce quattro tipi di creme, C_1, C_2, C_3, C_4, utilizzando 3 ingredienti base A, B e C. Per la produzione della crema C_4, inoltre, sono impiegate anche una certa quantità di C_1 e di C_2. In tabella sono riportate le quantità, in Kg, di componenti che sono necessari per produrre un Kg di ogni tipo di crema.

Crema	A	B	C	C_1	C_2
C_1	0,1	0,4	0,3	-	-
C_2	0,4	0,2	0,2	-	-
C_3	0,2	0,5	0,1	-	-
C_4	0,1	0,1	0,2	0,1	0,3

Per il prossimo mese sono stati acquistati 900, 1100 e 800 Kg di A, B e C, rispettivamente. Nella tabella seguente sono riportati i profitti (in Euro per Kg di prodotto) di vendita per ogni tipo di crema.

	C_1	C_2	C_3	C_4
Profitto	4	3	2,5	3,5

Formulare il problema di pianificare la produzione giornaliera in modo da massimizzare il profitto, sapendo che la quantità di crema C_4 prodotta non deve essere superiore a 500Kg.

Problema 3 (4 punti):

Trovare, nel seguente array di interi, un sottointervallo di interi consecutivi la somma dei cui valori sia **MINIMA**.

-18	8	-23	16	-20	42	-30	20	-24	33	-16	28	-7	15	-25	20	-7	8	-21	25	-13	17	-8	6	-6	4	-6

tipo intervallo	min sum	parte da pos.	arriva a pos.	parte da val.	arriva a val.
qualsiasi					
include ultimo					
include 9^o					
include 5^o e 10^o					
include 17^o					
include 14^o e 16^o					

Problema 4 (4 punti):

Si consideri la seguente sequenza di numeri naturali.

54	51	49	54	50	34	27	3	51	30	2	47	24	36	9	7	46	48	54	30	5	42	25	13	44

5.1(1pt) trovare una sottosequenza decrescente che sia la più lunga possibile. Specificare quanto è lunga e fornirla.

5.2(1pt) una sequenza è detta una S-sequenza, o sequenza decrescente con un possibile ripensamento, se esiste un indice i tale cha ciascuno degli elementi della sequenza esclusi al più il primo e l'i-esimo sono strettamente minori dell'elemento che immediatamente li precede nella sequenza. Trovare la più lunga S-sequenza che sia una sottosequenza della sequenza data. Specificare quanto è lunga e fornirla.

5.3(1pt) trovare la più lunga sottosequenza decrescente che includa l'elemento di valore 46. Specificare quanto è lunga e fornirla.

5.4(1pt) una sequenza è detta una A-sequenza se su una prima parte è crescente e poi sempre decrescente. Trovare la più lunga A-sequenza che sia una sottosequenza della sequenza data. Specificare quanto è lunga e fornirla.

Problema 5 (6 punti):

6.1**(1pt)** Costruire un problema di PL che sia illimitato.

6.2**(1pt)** Costruire un problema di PL che abbia infinite soluzioni ottime.

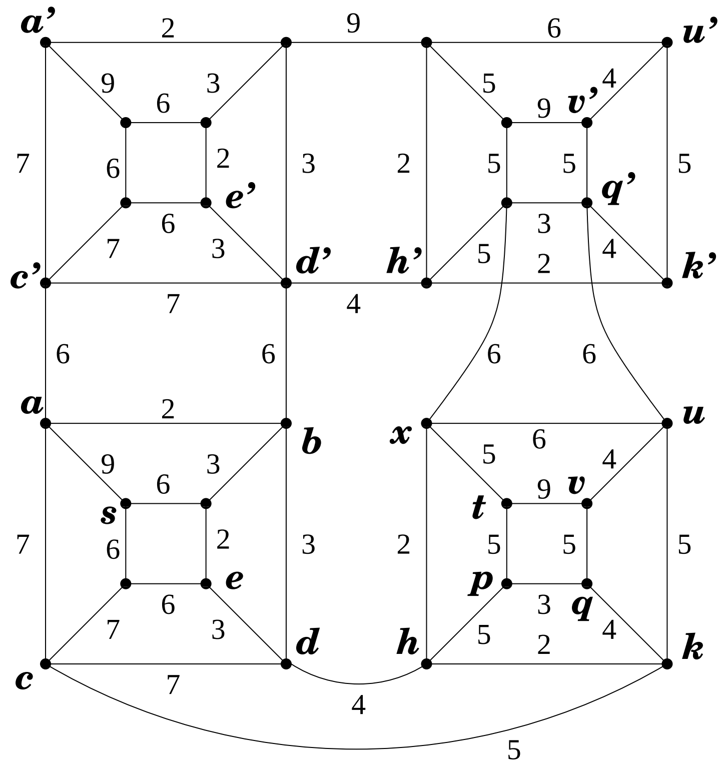
6.3**(1pt)** Costruire un problema di PL il cui duale abbia infinite soluzioni ottime.

6.4**(1pt)** Costruire un problema di PL che abbia precisamente 2 soluzioni ottime di base.

6.5**(1pt)** Costruire un problema di PL che abbia precisamente 3 soluzioni ottime di base.

6.6**(1pt)** Costruire un problema di PL in cui l'unica soluzione ottima sia degenere.

Problema 6 (15 punti):
Si consideri il grafo, con pesi sugli archi, riportato in figura.



7.1.**(2pt)** Trovare un albero ricoprente di peso minimo.

7.2.**(2pt)** Trovare tutti gli alberi ricoprenti di peso minimo. (Dire quanti sono e specificare con precisione come generarli).

7.3.**(3pt)** Trovare un massimo flusso dal nodo s al nodo t.

7.4.**(2pt)** Certificare l'ottimalità del flusso massimo dal nodo s al nodo t.

7.5.**(1pt)** Dire, certificandolo, se il grafo è bipartito oppure no.

7.6.**(2pt)** Se il grafo è bipartito, aggiungere un arco in modo che esso cessi di essere bipartito. In caso contrario, dire quale è il minimo numero di nodi la cui rimozione rende il grafo bipartito. (Fornendo non solo certificato di bipartizione per il grafo ottenuto ma anche argomentando che la rimozione di un numero minore di archi non può bastare).

7.7.**(2pt)** Dire, certificandolo, se il grafo è planare oppure no.

7.8.**(1pt)** Se il grafo è planare, aggiungere un arco in modo che esso cessi di essere planare. In caso contrario, dire quale è il minimo numero di archi la cui rimozione rende il grafo planare. (Fornendo certificato di planarità per il grafo ottenuto ed argomentando che la rimozione di un numero minore di archi non può bastare).