NOME: .... COGNOME: ....

MATRICOLA: .... FIRMA: ....

Esame di Ricerca Operativa - 13 giugno 2012
Facoltà di Ingegneria - Udine

Problema 1 (4 punti):

La fonderia Beldur produce un acciaio, ottenuto dalla fusione di 4 diversi materiali grezzi. Il costo unitario (Euro/kg) di ciascun materiale e la su composizione espressa in percentuali kg/kg di materiale, sono espressi nella seguente tabella:

	% alluminio	% silicio	% carbonio	costo al kg
materiale 1	2	9	7	700
materiale 2	5	8	7	600
materiale 3	3	6	5	500
materiale 4	4	6	7	650

Si tenga conto che il prodotto finale deve contenere una percentuale di alluminio compresa tra il 3% e l'8%, una percentuale di silicio tra il 4% e il 5%, e una percentuale di carbonio non superiore al 5%. Formalizzare il problema di pianificare la produzione della fonderia con l'obiettivo di minimizzare i costi.

Problema 2 (2+2 punti):

Data la formula booleana $(x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_3 \vee x_4 \vee \overline{x}_5) \wedge (\overline{x}_1 \vee x_2 \vee \overline{x}_4) \wedge (\overline{x}_2 \vee x_3 \vee \overline{x}_5) \wedge (\overline{x}_3 \vee x_5)$ siamo interessati a quegli assegnamenti di valori di veritá alle variabili che rendano vera la formula (ossia che soddisfino ciascuna delle sue 5 clausole).

Assumiamo tuttavia che settare la variabile x_i a *true* comporti dei costi come da seguente tabella.

	x_1	x_2	x_3	x_4	x_5
cost of truth	2	5	-3	4	-1

(2pt) Formulare come problema di programmazione lineare intera (PLI) l'intento di soddisfare la formula a costo minimo.

(2pt) Piú in generale, data una formula booleana in forma normale congiuntiva $\Phi = \wedge_{i=1}^{m} \left(\vee_{j=1}^{|p_i|} x_{p_i(j)} \right) \vee \left(\vee_{j=1}^{|n_i|} x_{n_i(j)} \right)$, ossia una disgiunzione (.OR.) di m clausole, dove p_i é un vettore che restituisce gli indici delle variabili che appaiono positive nella clausola i e $|p_i|$ indica la lunghezza di p_i, mentre n_i é un vettore che restituisce gli indici delle variabili che appaiono negate nella clausola i e, analogamente, $|n_i|$ é il numero di variabili che compaiono negate nella clausola i, si esprima come un problema di PLI la ricerca di assegnamenti di veritá che soddisfino alla formula a costo minimo, dove con c_j, $j = 1, 2, \ldots, n$, indichiamo il costo di settare la variabile x_j a *true*.

Problema 3 (4 punti):
Trovare la più lunga sottosequenza comune tra le stringhe $s =$ C T G T G A G A A T C G C T G T A e $t =$ G T A C G A C T G A A G C T A T. Fare lo stesso con alcuni prefissi di s e t.

3.1(1pt) quale è la più lunga sottosequenza comune tra s e t?

3.2 (1pt) e nel caso sia richiesto che la sottosequenza comune termini con 'C'?

3.3 (1pt) quale è la più lunga sottosequenza comune tra s e il prefisso $t_9 =$ G T A C G A C T G di t?

3.4 (1pt) quale è la più lunga sottosequenza comune tra t e il prefisso $s_8 =$ C T G T G A G A di s?

tipo di sottosequenza comune	lunghezza	sottosequenza
qualsiasi		
termina con 'C'		
tra s e t_9		
tra s_8 e t		

Problema 4 (4 punti):
Si consideri la seguente sequenza di numeri naturali.

17	18	16	44	32	15	52	27	4	9	11	50	48	21	33	10	55	27	6	54	30	23	7	3	8	6	13

4.1(1pt) trovare una sottosequenza decrescente che sia la più lunga possibile. Specificare quanto è lunga e fornirla.

4.2(2pt) una sequenza è detta quasi-decrescente, 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 sequenza quasi-decrescente che sia una sottosequenza della sequenza data. Specificare quanto è lunga e fornirla.

4.3(1pt) trovare la più lunga sottosequenza decrescente che includa l'elemento di valore 11. Specificare quanto è lunga e fornirla.

tipo sottosequenza	max lung	sottosequenza ottima
decrescente		
quasi-decrescente		
decrescente con 11		

Problema 5 (8 punti):

$$\max\ 6x_1 - 5x_2 - 3x_3 + 6x_4$$
$$\begin{cases} -4x_1 + 5x_2 - 3x_3 + x_4 \leq 5 \\ 2x_1 - 5x_2 + x_3 - x_4 \leq -5 \\ 2x_1 - x_2 + 2x_3 - x_4 \leq 4 \\ x_1, x_2, x_3, x_4 \geq 0 \end{cases}$$

5.1(1pt) Impostare il problema ausiliario.

5.2(2pt) Risolvere il problema ausiliario per ottenere una soluzione ammissibile di base al problema originario.

5.3(2pt) Risolvere il problema originario all'ottimo.

5.4(1pt) Quanto si sarebbe disposti a pagare per ogni unità di incremento per l'availability nei tre vincoli? (Per piccole variazioni.)

5.5(2pt) Fino a dove si sarebbe disposti a pagare tali prezzi ombra?

Problema 6 (6 punti):

Progettare un problema di PL in forma standard (od argomentare che esso non esista) tale che:

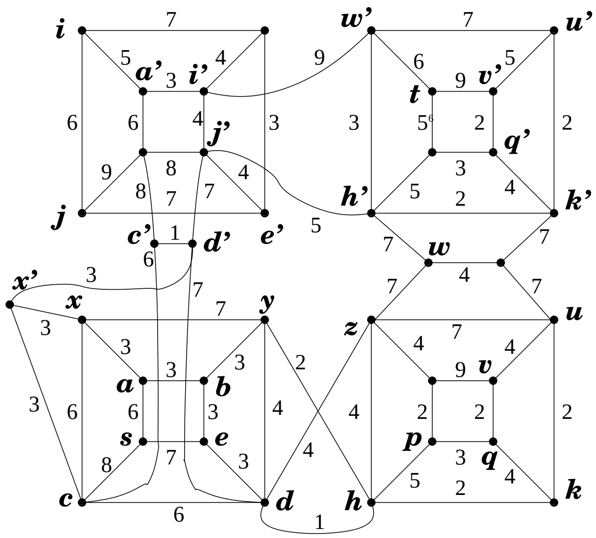
6.1 (2pt) ha esattamente 3 soluzioni di base ottime;

6.2 (2pt) ha infinite soluzioni ottime ma nessuna di esse é di base;

6.3 (1pt) il duale ha una soluzione degenere;

6.4 (1pt) il duale ha almeno 2 soluzioni di base ottime.

Problema 7 (18 punti):
Si consideri il grafo, con pesi sugli archi, riportato in figura.



7.1.**(1pt)** Dire, certificandolo, se il grafo è planare oppure no. In ogni caso, disegnare il grafo in modo da minimizzare il numero di incroci tra archi.

7.2.**(1+1pt)** Dire, certificandolo, se il grafo G' ottenuto da G sostituendo l'arco $c'c$ con un arco $c'x$ è planare oppure no. Se non planare, rimuovere il minimo numero di archi per planarizzarlo.

7.3.**(1+1+1pt)** Dire, certificandolo, se G e G' è bipartito oppure no. Ove non bipartito, rimuovere il minimo numero di archi per bipartizzarlo. (Certificando che quel numero di archi è necessario).

7.4.**(1+1pt)** Trovare l'albero dei cammini minimi dal nodo s. Esprimere la famiglia di tali alberi.

7.5.**(2pt)** Trovare un albero ricoprente di peso minimo.

7.6.**(2pt)** Trovare tutti gli alberi ricoprenti di peso minimo. (Dire quanti sono e specificare con precisione come generarli).

7.7.**(2pt)** Per i seguenti archi dire, certificandolo, in quale categoria ricadano (contenuti in ogni/nessuna/qualcuna-ma non-tutte le soluzioni ottime): ij, ab, $h'w$.

7.8.**(2pt)** Trovare un massimo flusso dal nodo s al nodo t.

7.9.**(2pt)** Certificare l'ottimalità del flusso massimo dal nodo s al nodo t.