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

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

Esame di Ricerca Operativa - 14 settembre 2010
Facoltà di Architettura - Udine

Problema 1 (1+1+2+1 punti):

Nel problema delle 8 regine viene chiesto di disporre 8 regine sulla scacchiera classica 8×8, stando attenti che non vi siano due regine mutualmente in presa (stessa riga o colonna, o che si vedano in diagonale). Il problema è stato studiato in generale per scacchiere $n \times n$. Ovviamente non è mai possibile disporre più di n regine. (Perchè? Un'argomento stringente ti vale il primo punto).

Ti sarà facile verificare che il problema ammette soluzione per $n = 0, 1$ ma non per $n = 2$. (Mentre nella scacchiera 1×1 è possibile disporre 1 regina, nella scacchiera 2×2 non è possibile disporre 2 regine). Quale è il più piccolo valore di $n \geq 2$ per il quale il problema ricomincia ad avere soluzioni? Questa sfida/gioco ti vale il secondo punto.

Formulare come un problema di Programmazione Lineare Intera (PLI) il caso di $n = 4$ ti vale due punti. Formulare come un problema di Programmazione Lineare Intera (PLI) il caso di n generico ti consegna l'ultimo punto.

Problema 2 (4 punti):

Un aereo da carico ha tre compartimenti per il trasporto delle merci: fronte, centro, coda. Vi sono dei limiti sul peso e volume di merce trasportabile nei tre compartimenti. La seguente tabella specifica tali limiti in megagrammi (tonnellate) ed in metri cubi, rispettivamente:

Compartimento	Peso (Mg)	Spazio (m^3)
Fronte	10	6800
Centro	16	8700
Cosa	8	5300

Inoltre, per garantire un assetto equilibrato all'aereo, il peso del carico deve essere ripartito sui tre compartimenti secondo le stesse percentuali delle capacità totali dei singoli compartimenti.

Per il prossimo volo abbiamo a disposizione le seguenti 4 tipologie di merce da carico.

Cargo	Peso (Mg)	Volume (m^3/Mg)	Profitto (Euro/Mg)
C_1	18	480	310
C_2	15	650	380
C_3	23	580	350
C_4	12	390	285

Una qualsiasi porzione di queste merci disponibili può essere trasportata (la tabella specifica solo la quantità massima, ossia quella attualmente presente nei magazzini di terra). Formulare come problema di programmazione lineare il problema di determinare quanto trasportare di ciascuna merce e come ripartirla sui compartimenti col fine di massimizzare il profitto.

Problema 3 (4 punti):
Un robot R deve portarsi dalla cella A–1 alla sua home H nella cella G–8.

	1	2	3	4	5	6	7	8
A	R	·	·	·	·	·	●	·
B	·	●	·	●	●	·	·	·
C	·	·	·	·	·	·	·	·
D	·	●	·	·	·	●	·	·
E	·	·	·	●	·	·	·	·
F	·	·	·	·	·	·	●	·
G	·	·	·	●	·	·	·	H

I movimenti base possibili sono il passo verso destra (ad esempio dalla cella A–3 alla cella A–4), il passo verso in basso (ad esempio dalla cella A–3 alla cella B–3), ed il passo obliquo basso/destra (ad esempio dalla cella A–1 alla cella B–2). Tuttavia il robot non può visitare le celle occupate da un pacman (●). Quanti sono i percorsi possibili?

2.1(1pt) Quanti sono i percorsi possibili se la partenza è in A–1?

2.2 (1pt) e se la partenza è in B–3?

2.2 (1pt) e se con partenza in A–1 il robot deve giungere in F–6?

2.4 (1pt) partenza in A–1 ed arrivo in G–8, al robot viene richiesto di passare per D–5.

consegna	numero percorsi
A–1 → G–8	
B–3 → G–8	
A–1 → F–6	
passaggio per D–5	

Problema 4 (4 punti):
Trovare la più lunga sottosequenza comune tra le stringhe $s = ATGTCAGAAGAGTCGTA$ e $t = GTACTGACTGAAGGTAT$. Fare lo stesso con alcuni suffissi 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 incominci con 'C'?

3.3 (1pt) quale è la più lunga sottosequenza comune tra s e il suffisso $t_9 = TGAAGGTAT$ di t?

3.4 (1pt) quale è la più lunga sottosequenza comune tra t e il prefisso $s^{14} = ATGTCAGAAGAGTC$ di s?

tipo di sottosequenza comune	lunghezza	sottosequenza
qualsiasi		
parte con 'C'		
tra s e t_9		
tra s^{14} e t		

Problema 5 (4 punti):
Si consideri la seguente sequenza di numeri naturali.

16	8	10	5	9	25	32	56	11	29	57	12	35	23	50	52	13	9	6	29	54	17	34	46	18

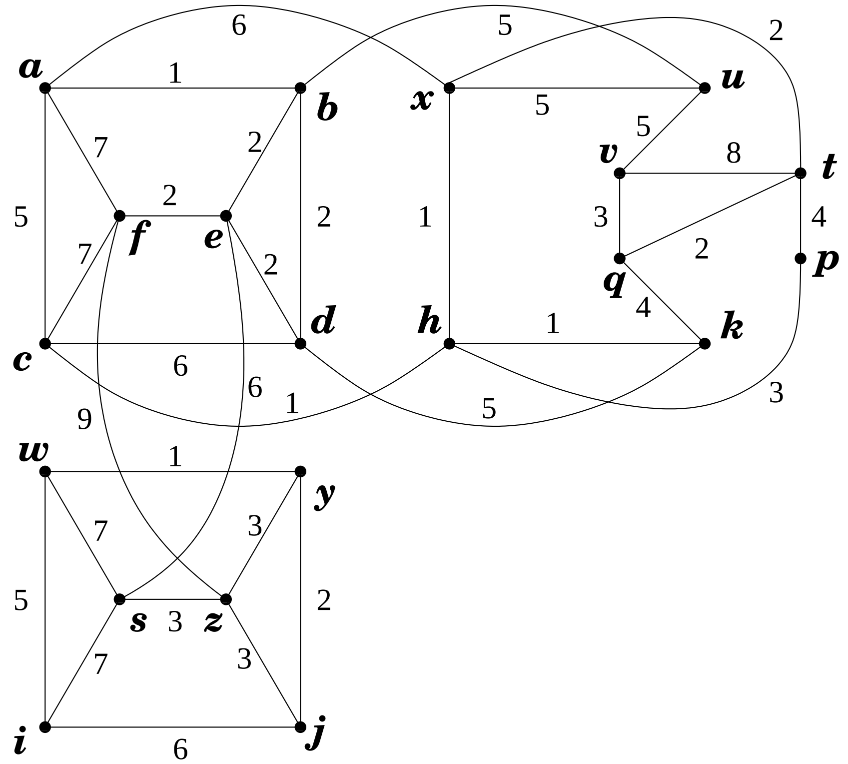
- **5.1(1pt)** trovare una sottosequenza crescente che sia la più lunga possibile. Specificare quanto è lunga e fornirla.
- **5.2(2pt)** una sequenza è detta una Z-sequenza, o sequenza crescente con un possibile ripensamento, se esiste un indice i tale cha ciascuno degli elementi della sequenza esclusi al più il primo e l'i-esimo sono strettamente maggiori dell'elemento che immediatamente li precede nella sequenza. Trovare la più lunga Z-sequenza che sia una sottosequenza della sequenza data. Specificare quanto è lunga e fornirla.
- **5.3(1pt)** trovare la più lunga sottosequenza crescente che includa l'elemento di valore 13. Specificare quanto è lunga e fornirla.

tipo sottosequenza	max lung	sottosequenza ottima
crescente		
Z-sequenza		
crescente con 13		

Problema 6 (6 punti):

- 6.1**(1pt)** É possibile costruire un problema di PL che sia illimitato ed il cui duale abbia infinite soluzioni ottime? Certificare la risposta.
- 6.2**(1pt)** É possibile costruire un problema di PL che sia non ammissibile ed il cui duale sia anch'esso non ammissibile? Certificare la risposta.
- 6.3**(1pt)** É possibile costruire un problema di PL che sia non ammissibile ed il cui duale abbia infinite soluzioni ottime? Certificare la risposta.
- 6.4**(1pt)** Costruire un problema di PL in forma standard il cui duale abbia infinite soluzioni ottime e precisamente 2 soluzioni ottime di base.
- 6.5**(1pt)** Costruire un problema di PL in forma standard con due variabili e precisamente 5 soluzioni ottime di base.
- 6.6**(1pt)** Costruire un problema di PL in forma standard la cui unica soluzione ottima sia degenere.

Problema 7 (16 punti):
Si consideri il grafo, con pesi sugli archi, riportato in figura.



7.1.**(1pt)** Dire, certificandolo, se il grafo è planare oppure no.

7.2.**(2pt)** Trovare un albero ricoprente di peso minimo.

7.3.**(2pt)** Trovare tutti gli alberi ricoprenti di peso minimo. (Dire quanti sono e specificare con precisione come generarli).

7.4.**(1pt)** Trovare un albero dei cammini minimi dal nodo s a tutti gli altri nodi del grafo.

7.5.**(2pt)** Dire, per ogni arco del grafo, se esso possa essere rimosso senza allontanare alcun nodo dal nodo s.

7.6.**(2pt)** Trovare un massimo flusso dal nodo s al nodo t.

7.7.**(2pt)** Certificare l'ottimalità del flusso massimo dal nodo s al nodo t.

7.8.**(1pt)** Dire, certificandolo, se il grafo è bipartito oppure no.

7.9.**(2pt)** Dire, certificandolo, quale sia il minimo numero di nodi la cui rimozione rende il grafo bipartito.

7.10.**(1pt)** Dire, certificandolo, se il grafo ottenuto aggiungendo l'arco di estremi v e p è planare oppure no.