

Elaborato sull'utilizzo del risolutore Gurobi

Suggerimenti

1. Per rispondere ai quesiti proposti, dovete sfruttare le conoscenze teoriche fino ad ora acquisite riguardanti la Programmazione Lineare e quelle che acquisirete riguardanti la Programmazione Lineare Intera.
2. Relativamente al codice che andrà scritto in Java:
 - 2.1 attenzione all'errore di macchina: potrebbe capitare che Gurobi vi calcoli un valore di una variabile pari a 0.9999999999; ciò vuol dire che, nella realtà, il valore di quella variabile è pari a 1;
 - 2.2 approssimare ogni valore calcolato alla quarta cifra decimale per arrotondamento;
 - 2.3 tenere presente che Gurobi definisce le variabili di surplus **negative**.

Istruzioni

1. Ogni risposta ai quesiti deve essere frutto di una o più linee di codice (**non è consentito svolgere calcoli "a mente", su carta o tramite altri software e poi semplicemente stampare a video le risposte**)
2. Potete utilizzare qualsiasi classe e metodo forniti dall'interfaccia Java di Gurobi (cfr. slide fornite in Comunità e documentazione ufficiale: *clicca qui per accedere alla documentazione*).
3. Il [codice sorgente](#) prodotto dovrà
 - consistere in una sola classe Java;
 - essere **debitamente commentato**, evidenziando, a grandi linee, le rispettive tre parti di codice che sono servite per rispondere ai tre quesiti;
 - stampare a video le risposte ai tre quesiti, secondo il formato descritto in Pagina 4.
4. Redigete una breve [relazione](#) (**non più di una pagina**) in formato pdf che riporti il modello risolto per il Quesito I, il modello risolto per il Quesito III e, in maniera concisa ma chiara, le metodologie adottate per svolgere ciascun sotto-quesito.
5. Non è possibile contattare il docente o gli assistenti per richieste relative alla parte teorica o alla stesura del codice, mentre è possibile chiedere eventuali chiarimenti inerenti alla consegna e all'interpretazione dei parametri nel file di testo.

CONSEGNA

La consegna è da completare **entro le 23:59 del 19 maggio 2024**. Devono essere caricati in Comunità Didattica, tramite l'oggetto "Consegna elaborato Gurobi", sia il codice sorgente Java che la relazione. In caso i parateri del problema vengano letti direttamente dal file di testo fornito (cfr. Pagina 4), caricare l'intera cartella del progetto Java posizionando il file di testo nel giusto directory. L'elaborato del gruppo di chi non avesse caricato tutto il materiale richiesto entro il tempo limite sarà considerato **insufficiente**.

Testo del problema (ICHEA)

Siete alla guida di ICHEA, un'impresa manifatturiera che produce n tipologie di cucine (insieme K), utilizzando m diversi materiali (insieme M). La lavorazione di ogni cucina richiede il passaggio attraverso l reparti (insieme D). Ogni specifica tipologia di cucina $i \in N$ richiede un certo tempo w_{ij} di lavorazione in ogni reparto $j \in D$ e una determinata quantità q_{iz} del materiale $z \in M$. Vi sono noti: il tempo t_j che ogni reparto $j \in D$ può complessivamente dedicare alla lavorazione delle cucine, la quantità massima complessiva Q_z a disposizione per ogni materiale $z \in M$, il prezzo di vendita unitario c_i (in euro) di ogni tipologia di cucina $i \in K$, il costo di produzione unitario p_i (in euro) di ogni tipologia di cucina $i \in K$. L'impresa vuole assicurarsi un potenziale profitto netto di almeno P euro. Quale è il piano di produzione ottimale (i.e., quante e quali cucine produrre) affinché sia minimo il tempo di inutilizzo complessivo nei reparti?

Nel file `.txt` allegato, rinominato con il nome del proprio gruppo, è contenuta la lista dei parametri dell'istanza del problema da studiare. In calce, è riportato un esempio del contenuto di questo file. **Si consiglia di scrivere un metodo in Java per leggere direttamente i dati dal file di testo.**

Quesiti

I Elaborare un modello matematico di Programmazione Lineare Intera \mathcal{M}_{PLI} per il problema fornito.

- a. Implementare e risolvere \mathcal{M}_{PLI} tramite Gurobi Optimizer, stampando a video il valore ottimo della funzione obiettivo e delle variabili (sia di slack/surplus che non).
- b. Risolvere \mathcal{M}_{PL} , rilassamento continuo di \mathcal{M}_{PLI} , e indicare:
 - quali sono all'ottimo i vincoli attivi;
 - se la soluzione ottima è degenere;
 - se la soluzione ottima è unica o no;
 - se l'ottimo coincide con quello di \mathcal{M}_{PLI} e, se no, dire se la relazione che intercorre tra i due valori è concorde con quanto studiato.

II Elaborare ulteriori vincoli lineari per imporre

- che ICHEA debba scegliere solo s tipologie di cucine da produrre;
- che la quantità minima di produzione per la cucina $i \in K$, se scelta, è di u_i pezzi.

Integrare questi vincoli nel modello \mathcal{M}_{PLI} per ottenere un nuovo modello \mathcal{M}_{PLI}^* . Risolvere quest'ultimo con Gurobi e stampare a video il valore ottimo della funzione obiettivo.

III Elaborare due metodi distinti in Java per

- stabilire e stampare la minima e massima variazione (Δ) che potrebbe assumere P (profitto minimo che l'azienda vuole ottenere) senza far variare la base ottima di \mathcal{M}_{PL} ;
- stabilire e stampare il massimo decremento intero che t_e (tempo disponibile del reparto e) potrebbe subire tale che il valore ottimo di \mathcal{M}_{PLI}^* rimanga quello trovato al quesito (II).

Esempio di file `.txt` commentato:

```
n 16
l 12
m 40
Q 401 910 ...
w
8 9 10 ...
.
.
.
7 14 ...
t 460 422 ...
q
15 20 ...
.
.
.
11 30 ...
c 4448 4792 ...
p 965 1487 ...
u 2 4 ...
P 14901
s 9
e 7
```

La riga etichettata con **n** indica il valore del parametro n .

La riga etichettata con **l** indica il valore del parametro l .

La riga etichettata con **m** indica il valore del parametro m .

La riga etichettata con **Q** presenta la lista degli m parametri Q_z . Ad esempio: 401 è la disponibilità del material 1, 910 è la disponibilità del materiale 2, e così via. Questa interpretazione è valida per ogni riga contenente una lista di parametri.

La riga etichettata con **w** introduce la matrice dei parametri w_{ij} , composta da n righe e l colonne. Nella riga i -esima sono indicati i tempi, in minuti, che la cucina i richiede in ognuno degli l reparti. Ad esempio: la cucina 1 necessita di 8 minuti di lavorazione nel reparto 1, 9 minuti nelreparto 2, e così via;

La riga etichettata con **t** presenta la lista degli l parametri t_j .

La riga etichettata con **q** introduce la matrice dei parametri q_{iz} , composta da n righe e m colonne. Nella riga i -esima sono indicate le unità che la cucina i richiede per ognuno degli m materiali.

La riga etichettata con **c** presenta la lista degli n parametri c_i .

La riga etichettata con **p** presenta la lista degli n parametri p_i .

La riga etichettata con **u** presenta la lista degli n parametri u_i .

La riga etichettata con **P** indica il valore del parametro P .

La riga etichettata con **s** indica il valore del parametro s .

La riga etichettata con **e** indica, indicizzato da 0, il valore del pedice e presente nel parametro t_e da decrementare.

La liste sono formate da una sequenza di numeri, tutti sulla stessa riga, separati da spazi. La cardinalità degli insiemi possono essere dedotte dal numero di elementi della lista. Dopo la parola-chiave che identifica il tipo di parametro descritto dalla riga, tutti i seguenti caratteri non bianchi sono numertici. Le matrici sono sempre introdotte dalla relativa parola-chiave subito seguita da un carattere a capo. Ogni riga della matrice ha sempre lo stesso numero di elementi. Il numero di righe della matrice è sempre noto a priori in base ad un altro parametro. Si consiglia di vedere la documentazione delle classi **Scanner** e **ArrayList**, in quanto possono fornire un utile supporto nella lettura del file di testo.

Esempio di output che il codice consegnato deve stampare a video:

```
GRUPPO <numero gruppo>
Componenti: <cognome componente 1> <eventuale cognome componente 2>

QUESITO I:
Obj intero = <valore funzione obiettivo>
<x> = <valore di x>
<y> = <valore di y>
<z> = <valore di z>
...
Obj rilassato = <valore funzione obiettivo>
Elenco nomi vincoli attivi: ...
Degenerare: sì|no
Multipla: sì|no
Rilassato coincide con Intero: sì|no

QUESITO II:
Obj nuovo intero: <valore funzione obiettivo>

QUESITO III:
intervallo DELTA P = [a, b] (#)
valore minimo t_e = ...
```

(#) stampare l'intervallo con estremi inclusi; se a o b risultasse $+\infty$ o $-\infty$, stampare +INF o -INF e modificare le parentesi.

NB: le risposte non stampate a video verranno considerate **in bianco** e quindi valutate negativamente.