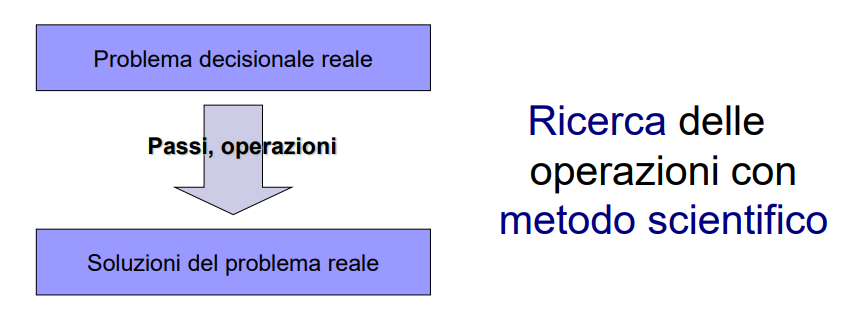
04/10/2022: Introduzione al corso

Ricerca Operativa (Operations Research) 🡪 Supporto ai processi decisionali in sistemi complessi, operando la decisione migliore a seconda del tipo di problema decisionale.

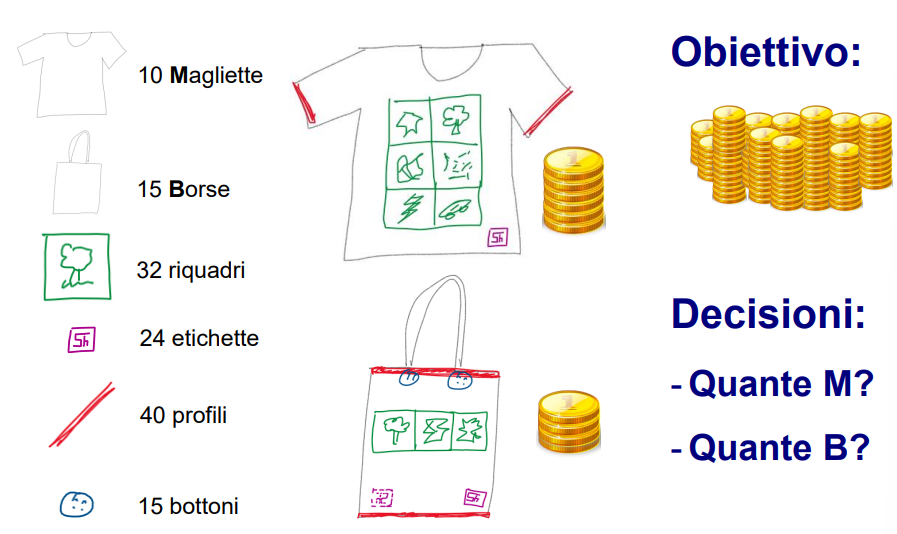
Cerco i passi che mi permettono di passare dalla decisione del problema alla soluzione stessa, applicando dei passi (quindi, un algoritmo), cercando un’ottimizzazione (soluzione migliore possibile)



Da Wikipedia, definizione espansiva e quella presente alla data di scrittura di questo file:

*“La ricerca operativa (nota anche come teoria delle decisioni, scienza della gestione o, in inglese, operations research ("Operational Research" in Europa) e indicata con le sigle RO o OR) è la branca della matematica applicata in cui problemi decisionali complessi vengono analizzati e risolti mediante modelli matematici e metodi quantitativi avanzati (ottimizzazione, simulazione, ecc.) come supporto alle decisioni stesse. La ricerca operativa riveste un ruolo importante nelle attività decisionali perché permette di operare le scelte migliori per raggiungere un determinato obiettivo rispettando vincoli che sono imposti dall'esterno e non sono sotto il controllo di chi deve compiere le decisioni.*

*L'obiettivo è dunque quello di fornire un supporto alla presa di decisioni. Per giungere a questo scopo, la ricerca operativa fornisce strumenti matematici e algoritmici di supporto alle attività decisionali in cui occorre gestire e coordinare attività e risorse limitate al fine di massimizzare o minimizzare una funzione obiettivo.*

*La ricerca operativa si occupa dunque di formalizzare un problema in un modello matematico e calcolare una soluzione ottima, quando possibile, o approssimata (detta anche subottima) per esso. Essa costituisce un approccio scientifico alla risoluzione di problemi complessi, si può ricondurre all'ambito della matematica applicata, ma presenta forti caratteristiche interdisciplinari relative in prevalenza a matematica, informatica, economia e finanza, ingegneria ed altre. Inoltre, la ricerca operativa ha molte applicazioni commerciali soprattutto negli ambiti economico, infrastrutturale, logistico, militare, della progettazione di servizi e di sistemi di trasporto e nelle tecnologie. Nel caso particolare di problemi di carattere economico, la funzione da massimizzare può coincidere con il massimo profitto ottenibile o con il minor costo da sostenere.”*

Esempio pratico di ottimizzazione: creazione di magliette e borse con un certo numero di riquadri.

L’obiettivo sarebbe di creare un programma per massimizzare il profitto date queste risorse.

Serve un *criterio* per capire come ragionare sulle risorse. Per esempio, posso ragionare ordinando in base al profitto, sommando le risorse che si devono usare e dividendo per quantità.

Altri ragionamenti possibili:

* scelte greedy (migliori sulla base di un input) sulla base delle magliette e delle borse
* bruteforce/approccio esaustivo-esponenziale (provo tutte le combinazioni possibili)
* scelgo una quantità casuale e cerco di vedere quale tra tutte si avvicina di più al valore che mi serve

Problemi di ottimizzazione

Determinare la migliore configurazione di sistemi complessi sotto condizioni di utilizzo di risorse scarse

* Mix ottimo di produzione
* Pianificazione della produzione, schedulazione di processi
* Determinazione dei turni del personale
* Determinazione di percorsi ottimali
* Organizzazione dei flussi di dati in una rete di telecomunicazione
* Individuazione di sequenze genomiche
* Pianificazione e gestione operativa di reti di trasporto (Amazon)

Si cerca in effetti di:

* Generare delle soluzioni ammissibili, quindi valide nel contesto d’uso
* Proporre delle soluzioni ragionevoli, quindi contestualmente utili nel contesto d’uso

Esempi utili e reali in merito (approfondimenti messi a scopo didattico)

* *Problema del commesso viaggiatore – Traveling salesman problem*

Il problema è NP-Completo, quindi un problema che è possibile verificare in tempo polinomiale (cioè, verificare la validità dell’ipotesi), ma non risolvibile nello stesso tempo, risultando un problema complesso.

Ci troviamo di fronte ad un quesito assai naturale: data una serie di città, qual è il modo più veloce (ovvero la strada più corta) per un commesso viaggiatore di visitare tutte le città una e una sola volta, e ritornare da dov’è partito?

È un problema complesso da risolvere; al crescere del numero di possibili città, il problema potrebbe non avere mai soluzione. Una soluzione venne dal prof. Christofides, il quale creò l’albero più corto che connette tutte le città, rete che connette tutte le città senza creare cicli. Se il percorso esiste, ogni città deve avere un numero pari di strade che la raggiungono. n altre parole, per ogni città, ogni arrivo è seguito da una partenza, il che significa che ci serve un numero pari di archi che tocchino la città nel network che costruiamo. La buona notizia è che anche l’inverso è vero: in un network in cui ogni città ha un numero pari di strade che la toccano, esiste un percorso che passa per ogni città una e una sola volta e torna alla città di partenza.

L’algoritmo, partendo dal network senza cicli, va ad aggiungere archi alle città con un numero dispari di strade, fintanto che ne esistono. Il percorso che ne esce alla fine non sarà necessariamente il migliore che il nostro commesso viaggiatore possa scegliere, ma non si allontana troppo dalla migliore scelta. Infatti, Christofides dimostrò che il suo metodo crea nel peggiore dei casi un percorso che è al massimo il 50% più lungo del percorso ottimo.

Esiste un nuovo algoritmo trovato dopo circa 50 anni, comunque molto simile a quello di Christofides. La differenza fondamentale consiste nel fatto che l’albero da cui si parte non è più il più corto possibile, ma è creato per via di un processo probabilistico e facendo in modo che le città con un numero dispari di strade siano vicine tra loro. Dopodiché l’algoritmo riprende l’originale e aggiunge strade a queste città fino ad ottenere un loop completo.

* Problema dello zaino – Knapsack problem

Nel problema dello zaino, è necessario imballare un insieme di oggetti, con valori e dimensioni date (come pesi o volumi), in un contenitore con una capacità massima. Se la dimensione totale degli oggetti supera la capacità, non è possibile imballarli tutti. In questo caso, il problema è scegliere un sottoinsieme di oggetti di valore totale massimo che possa entrare nel contenitore.

Esistono due tipi di problemi knapsack:

* 0/1 Knapsack Problem

Significa che gli oggetti o sono completamente o non sono riempiti in uno zaino (quindi, o li metti interi o non li metti). Ad esempio, abbiamo due oggetti che pesano rispettivamente 2 kg e 3 kg. Se prendiamo l'oggetto da 2 kg, non possiamo prendere un oggetto da 2 kg (l'oggetto non è divisibile); dobbiamo prendere completamente l'oggetto da 2 kg.

Il problema si risolve con la programmazione dinamica. Per risolverlo con la programmazione dinamica, prendiamo pesi e valori degli oggetti, cercando di assegnare, risolvendo i sottoproblemi in una matrice unica (quello che fa la programmazione dinamica), tenendo conto delle assegnazioni di quantità e pesi rispettive nelle varie applicazioni.

In questo modo, sappiamo caso per caso se riusciamo ad assegnare pesi e quantità.

* Fractionary knapsack problem

Significa che possiamo dividere l'oggetto. Ad esempio, se abbiamo un oggetto di 3 kg, possiamo prendere l'oggetto di 2 kg e lasciare quello di 1 kg. Il problema dello zaino frazionato si risolve con l'approccio Greedy. L'idea di base dell'approccio greedy consiste nel calcolare il rapporto valore/peso per ogni elemento e nell'ordinare gli elementi in base a questo rapporto. Quindi si prende l'elemento con il rapporto più alto e lo si aggiunge fino a quando non è possibile aggiungere l'elemento successivo nella sua interezza e alla fine si aggiunge l'elemento successivo il più possibile.

Però:

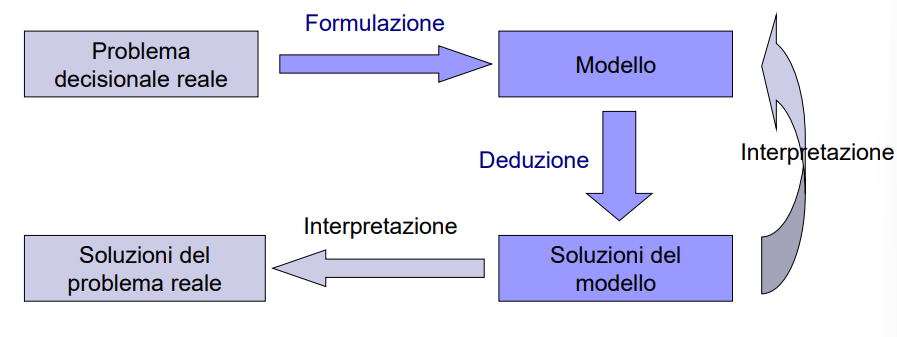
◼ Come certificare che una soluzione proposta è la migliore in assoluto (ottima)?

◼ Come valutare il valore intrinseco (significato qualitativo) delle risorse (e.g. un riquadro)

◼ Come valutare la stabilità della soluzione proposta in funzione di variazioni dei dati (ricavi, risorse disponibili etc.)?

◼ Come stabilire le soluzioni ottime in problemi simili (prospettiva modellistica e algoritmica)?

Per risolvere questo, si ha l’uso di strumenti matematici e algoritmici (teorema e dimostrazione), tramite la Ricerca Operativa, certificando la soluzione ottima.

Il metodo della Ricerca Operativa è basato su:

◼ Formulazione: modelli matematici, su grafo, di simulazione, teoria dei giochi, data-driven

(intelligenza artificiale)

◼ Deduzione: metodi quantitativi, algoritmi efficienti.

Nell’uso di un modello, non si trova una soluzione del problema; questo ovviamente ha dei limiti, in quanto la soluzione deve essere interpretata sulla base della realtà (se è un buon modello, si adatta bene).

Vediamo un esempio reale per capire l’utilizzo della matematica:

*“Un coltivatore ha a disposizione 11 ettari di terreno da coltivare a lattuga o a patate. Le risorse a sua disposizione, oltre al terreno, sono: 70 kg di semi di lattuga, 18 t di tuberi, 145 mc di fertilizzante. Supponendo che il mercato sia in grado di assorbire tutta la produzione e che i prezzi siano stabili, la resa stimata per la coltivazione di lattuga è di 3000 €/ettaro e quella delle patate è di 5000 €/ettaro. Il consumo di risorse per ogni tipo di coltivazione è di 7 kg di semi e 10 mc di fertilizzante per ettaro di lattuga, e 3 t di tuberi e 20 mc di fertilizzante per le patate. Stabilire quanto terreno destinare a lattuga e quanto a patate in modo da massimizzare la resa economica e sfruttando al meglio le risorse disponibili.”*

Dati utili:

* Lattuga e Patate (quali risorse)
* Semi, Tuberi e Fertilizzante (quali sottotipi di risorse e incrocio possibile dei dati)

Di fatto, in pratica, è come il problema di prima.

Non si può adottare un approccio forza bruta (quantità continue e non discrete). In realtà, questo ci aiuta.

*Costruzione del modello*

◼ Cosa bisogna decidere? ⇒ variabili decisionali (incognite che rappresentano i dati)

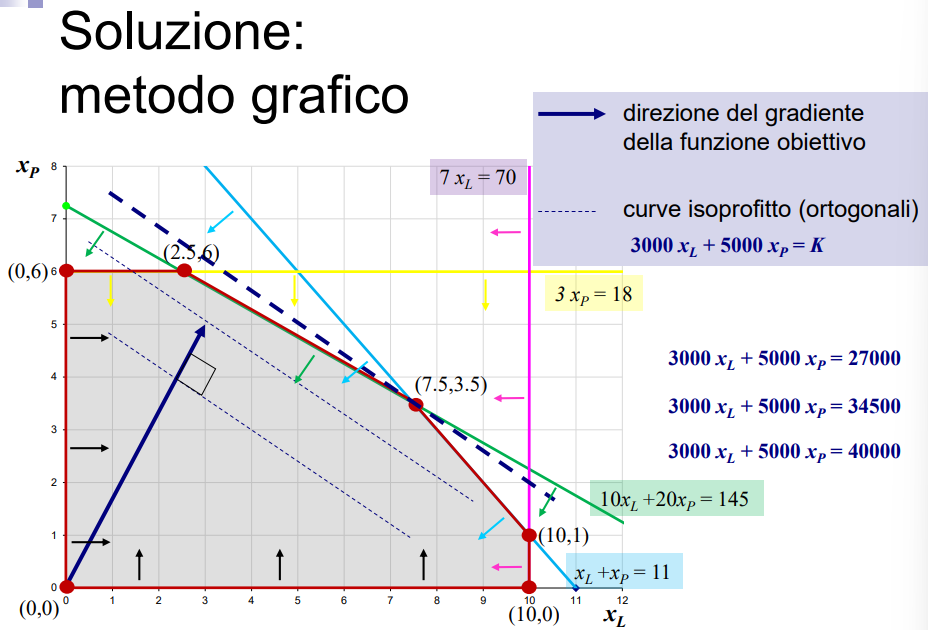
◼ Qual è l’obiettivo? ⇒ funzione obiettivo (ciò che vogliamo ottenere, mette insieme le v. decisionali)

◼ Come sono caratterizzate le soluzioni ammissibili? ⇒ vincoli del problema (relazioni tra incognite)

Immagine che contiene testo

Descrizione generata automaticamente◼ Modelli matematici ⇒ f. obiettivo e vincoli visti come relazioni matematiche tra le variabili decisionali

Aggiungiamo che quelle citate sotto come curve isoprofitto sono la differenza tra il ricavo ed i costi totali.



Per trovare la soluzione ottima, usiamo la funzione obiettivo.

Man mano che aumenta il guadagno, ho sempre più rette che vanno verso il *gradiente* (piano delle soluzioni ammissibili in rosso, possiamo dire il vettore che descrive il “movimento” delle derivate parziali delle rette); finché ci sono rette parallele, aumento il profitto, fino ad arrivare al limite della regione.

La soluzione ottima qui è 40000, in cui la soluzione ottima è il punto (7.5, 3.5).

Il metodo grafico funziona grazie a due motivi:

⇒ la linearità della funzione obiettivo

⇒ la linearità dei vincoli

Si parla in questi casi di modelli di programmazione lineare (PL) (come in questo caso, almeno due variabili e funzioni isoprofitto sotto forma di rette).

Sotto queste ipotesi (come vedremo meglio in seguito), una soluzione si trova su un vertice della regione ammissibile: l’ultimo toccato traslando le rette isoprofitto nella direzione del gradiente.

Se ci sono più variabili, si cerca di passare ad un problema a livello geometrico, tale che si possa approssimare comunque algebricamente il problema.

*Programma di massima del corso:*

1. Problemi di ottimizzazione: modellazione e soluzione con software off-the-shelf (app già pronte da usare)

- formulazione di modelli di programmazione matematica;

- soluzione con l’utilizzo di pacchetti software (laboratorio).

2. Programmazione lineare:

- teoria e metodo del simplesso;

- teoria della dualità e applicazioni.

3. Ottimizzazione su grafi: modelli e algoritmi per

- problema del cammino minimo;

- problemi di flusso su reti (flusso massimo, flusso di costo minimo).

4. Introduzione alla Programm. Lineare Intera e all'Ottimizzazione Combinatoria

- metodo del Branch & Bound per PL;

- cenni sui metodi euristici e metaeuristici (ricerca locale e varianti)

05/10/2022: Modelli di programmazione lineare

Diamo quindi esplicitamente il concetto di modello di programmazione lineare, quindi una relazione che esprime la soluzione ottima di un problema di ottimizzazione attraverso relazioni matematiche lineari

(altri non lineari esistono, ma il metodo di soluzione sfrutta fortemente la linearità dei vincoli, sapendo che lo spazio definito come modello è un poliedro).

Gli elementi di un modello di programmazione sono:

• *Insiemi*: raggruppano gli elementi del sistema;

• *Parametri*: sono i dati del problema e rappresentano delle quantità fissate che dipendono dai diversi elementi del sistema;

• *Variabili decisionali o di controllo*: sono le grandezze del sistema di cui non conosciamo il valore (assimilabili a delle incognite) e sulle quali possiamo agire per determinare diverse soluzioni alternative del problema;

• *Vincoli (incognite)*: sono delle relazioni matematiche che descrivono le condizioni di ammissibilità delle soluzioni. Servono quindi per discriminare le combinazioni di valori delle variabili decisionali che rappresentano soluzioni accettabili del problema, da quelle che non lo sono;

• *Funzione obiettivo (relazione sulle incognite)*: è la quantità da massimizzare o minimizzare, espressa come funzione delle variabili decisionali.

Un modello di programmazione matematica *dichiara* le caratteristiche della soluzione cercata (che cosa), piuttosto che definire la strategia per la ricerca della soluzione stessa (*come*).

Per esempio, nel caso del modello della lattuga e patate, si sta dicendo:

* Qual è la soluzione ottima tale che ogni vincolo sia rispettato, ottenendo il valore massimo possibile (non dicendo *come* la si trova)

Nell’esercizio di modellazione, attenzione a non inserire un modello non lineare (errore grave).

Quella che segue è la formulazione (con le disuguaglianze che sono sempre o e non segni stretti, in quanto pericoloso da un punto di vista computazionale):

Immagine che contiene testo

Descrizione generata automaticamente

Si noti che intende “sotto il vincolo” oppure “tale che” (visto come “such that”)

*Programmazione lineare intera (PLI), quindi tutte le variabili possono assumere valore interi*

*Costanti note*

I modelli di programmazione lineare sono una particolare classe di modelli di programmazione matematica in cui:

• la funzione obiettivo è un’espressione lineare delle variabili decisionali;

• i vincoli sono determinati da un sistema di equazioni e/o disequazioni lineari.

Le variabili, di qualsiasi natura essere siano, possono essere solo moltiplicate per una costante e sommate tra loro. Si richiedono solo modelli lineari nel corso. Privilegiamo la semplicità del modello per velocizzare la risoluzione. Vediamo degli esempi di modellazione:

Immagine che contiene testo

Descrizione generata automaticamente1) Gioco di assemblaggio (telecomandi)

di 4

6

Quello che si vuol fare è esprimere le variabili decisionali. Quindi:

* # (numero) telecomandi tipo A,
* # telecomandi tipo B,

Si vuole massimizzare la quantità da 4 euro per A e da 6 euro per tipo B, quindi:

Importante esprimere il dominio delle variabili, quindi:

(dominio)

Quindi, dati i profitti unitari, vogliamo fare in modo che, per le singole risorse, non si superi la disponibilità:

Immagine che contiene testo, dispositivo

Descrizione generata automaticamente

Sapendo che, ad esempio:

Le variabili devono essere intere e positive

Si può risolvere il problema graficamente? Sì (esempio in Excel del prof), grazie al fatto che il vertice ottimo è intero.

Riprendiamo l’esempio delle magliette, esprimibile anch’essa come modello di PLI.

Quando si va nella direzione del gradiente, quando il modello è rappresentato graficamente, va nella direzione del vertice ottimo, il quale deve essere l’ultimo punto toccato dalle linee di livello all’interno della zona della soluzione ammissibile (sapendo qual è l’ultimo punto toccato; esso, però, deve essere intero).

Con Excel, per esempio, si può usare la funzione *SOLVER* che trova la soluzione ottima.

Quando si opera su due dimensioni, è più difficile, perché la soluzione ottima non sempre è vicina a dove siamo (essendo la soluzione intera, è un “punto che sta in mezzo” e non si riesce a caratterizzarlo bene, quindi complicando la risoluzione).

2) Dieta economica (non è importante dare una quantificazione temporale; il modello funziona lo stesso, non essendo importante il periodo per la caratterizzazione del problema. Più di una persona ha sollevato il dubbio che dovesse essere scritto “settimanale”; in realtà, non è importante).

Immagine che contiene testo

Descrizione generata automaticamente

Vincoli: e (non serve siano intere, in quanto anche qui la quantità proteica non può, logicamente, essere frazionaria; non mi serviranno 5,67 mg. di proteine, in concreto).

Una guida, spesso, nella modellazione è la funzione obiettivo.

(Eventualmente anche, per chiarificare con il contesto, possiamo chiamarle )

In questo caso, dobbiamo superare le soglie delle proteine, del ferro e del calcio (quindi )

Immagine che contiene testo

Descrizione generata automaticamente

Esempio esteso calcolo proteine:

Che diventa:

Immagine che contiene testo, antenna

Descrizione generata automaticamente

Si può risolverlo anche questo con il metodo grafico, risultando un piano nello spazio.

Se al posto di avere il fruttivendolo di quest’ipotesi, avessimo delle scatolette:

* Vanno aggiunte delle costanti moltiplicative per il peso dei barattoli e si aggiunge un vincolo intero alle variabili; quindi, si avrebbe ad esempio

06/10/2022: Esercizi di Programmazione Lineare

Riportiamo l’esercizio (3*), Indagine di mercato*:

Immagine che contiene tavolo

Descrizione generata automaticamente

Le variabili decisionali sono:

* , telefonate al mattino
* , telefonate alla sera

Il dominio è sugli interi positivi, dunque:

L’obiettivo è di minimizzare il costo delle telefonate, dunque la funzione obiettivo:

affinché:

* per le donne sposate
* per le donne non sposate
* per gli uomini sposati
* per gli uomini non sposati

Come teniamo conto dei nessuno (le chiamate a vuoto)?

Non abbiamo un caso migliore o peggiore delle telefonate, infatti non si ha controllo rispetto a quante vadano effettivamente a vuoto. Esse sono una “seccatura” già compresa nei vincoli, in quanto sono una percentuale di telefonate che non contribuisce a nulla.

Altro esercizio 🡪 (4), *Trasporto di frigoriferi*

Immagine che contiene testo

Descrizione generata automaticamente

Variabili decisionali:

: # frigo da a , ,

Funzione obiettivo 🡪 costo minimo

Immagine che contiene testo

Descrizione generata automaticamente

Quanto visto fino ad ora sono problemi “classici”, cosiddetti modelli di mix ottimo di produzione.

Immagine che contiene testo

Descrizione generata automaticamente

Altri problemi sono quelli di copertura di costo minimo (problemi di dieta):

Immagine che contiene testo

Descrizione generata automaticamente

Altro problema classico di problemi di trasporto:

Immagine che contiene testo

Descrizione generata automaticamente

Su un esempio concreto con indici rispetto a quest’ultimo modello:

e

Avremmo variabili decisionali e vincoli .

*Domanda*: Non avrebbe più senso mettere il segno “” e non “” o “”?

*Risposta*: In realtà no, poiché sarebbe una quantità *al limite*, dunque approssimativamente arriviamo alla quantità detta. Normalmente, questo si intende dai vincoli del testo o, comunque, dalla natura del problema. Se necessario si mette, altrimenti non serve.

*Altra domanda*: Potrà capitare una situazione in cui uno dei vincoli è rispettato con il maggiore stretto “>”?

*Risposta*: È ammissibile come soluzione, ma dipende dalla natura in problema.

Nel modello sopra (trasposto), non è ammissibile in quanto non è un problema di minimo/ottimo; in questo caso, quindi, non esiste soluzione che rispetti tale vincolo.

Vediamo un altro problema (5), *Turni in ospedale*:

Immagine che contiene testo

Descrizione generata automaticamente

*Variabili decisionali*

Sapendo che i miei turni sono compresi per singolo giorno, impieghiamo , idealmente abbiamo:

* gli infermieri che lavorano il lunedì
* gli infermieri che lavorano il martedì,
* gli infermieri che lavorano il mercoledì
* gli infermieri che lavorano il giovedì
* gli infermieri che lavorano il venerdì
* gli infermieri che lavorano il sabato
* gli infermieri che lavorano la domenica

Immagine che contiene tavolo

Descrizione generata automaticamente

(Si può, volendo, commentare come sul codice anche in esame, ad esempio come un commento alla C++, quindi // )

: # infermieri del turno -esimo

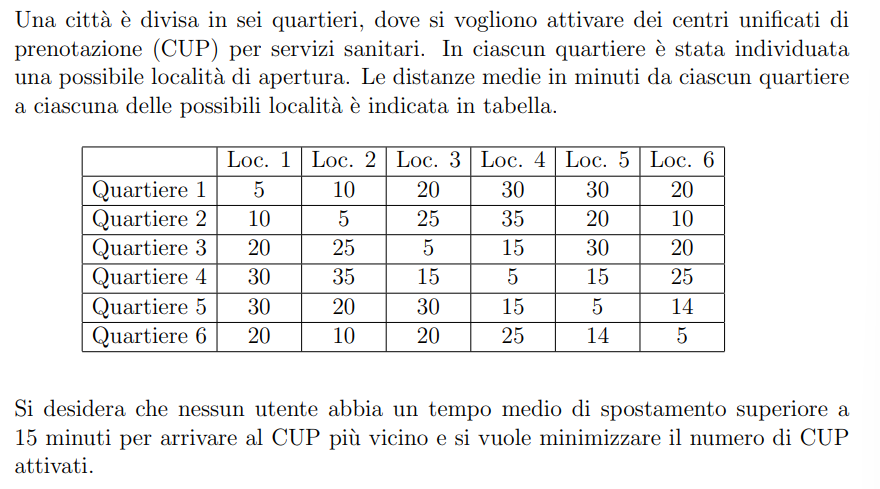
Esempio del lunedì 🡪

Esempio del martedì 🡪

Dominio 🡪

Si noti che il testo contiene, più o meno volutamente, delle ambiguità (caso dei segni di uguaglianza rispetto ai segni di disuguaglianza). Probabilmente, il modello darà come soluzione impossibile (nessuna soluzione ammissibile). Se una soluzione funzionasse, tra le due, si tende a preferire una o l’altra notazione opportunamente giustificata.

Altro esercizio (6), *Localizzazione di servizi*



Il problema *somiglia* a quello di copertura di costo minimo.