Pianificazione e Gestione Servizi Sanitari

Seminario CPLEX – pt2

Prof. Cristian Belfiore – Lezione 21 – 15/11/23 – Autori/Revisionatori: Salvati e Luciani

---

Riprendendo il problema della lezione precedente, il vincolo sugli stabilimenti da attivare traduce simultaneamente una condizione logica ed un vincolo tecnologico.

$$\sum_{i \in P} t_{ij} x_{ij} \leq T_j y_j \quad \forall j \in S$$

Il vincolo che ne viene fuori è $2x_{1,1} + 3x_{2,1} + x_{3,1} <= 10\ y_1$

Se $y_1 = 1$, lo stabilimento 1 viene attivato: in questo è possibile produrre degli articoli soddisfacendo il vincolo di disponibilità temporale.

Il vincolo che ne risulta è:

$x_{1,1} + x_{2,1} + x_{3,1} >= 0$ ma tali per cui $2x_{1,1} + 3x_{2,1} + x_{3,1} <= 10\ y_1$

Viceversa, se y1 = 0, lo stabilimento 1 non viene attivato: all'interno di questo non può essere prodotto nulla.

Il vincolo che ne risulta è:

$x_{1,1} + x_{2,1} + x_{3,1} = 0$ e cioè tutte e tre le variabili saranno nulle.

Medesimo risultato si otterrebbe scrivendo la seguente coppia di gruppi di vincoli:

$$\sum_{i \in P} t_{ij} x_{ij} \leq T_j \quad \forall j \in S$$

→ traduzione del vincolo tecnologico sulla disponibilità oraria degli stabilimenti

$$\sum_{i \in P} t_{ij} x_{ij} \leq \boxed{M}\, y_j \quad \forall j \in S$$

→ traduzione condizione logica
Nel vincolo precedente, invece di T si utilizza la costante M.
Big M = parametro sufficientemente grande da non costituire una limitazione aggiuntiva non voluta nel caso in cui lo stabilimento j venga attivato ($y_j = 1$)

PROBLEMA 5

Una grande Azienda Farmaceutica è specializzata nella produzione di un nuovo farmaco per pazienti affetti da scompenso cardiaco cronico.

Dal punto di vista della distribuzione, il farmaco viene fornito ai centri di smistamento in modo periodico, alla fine di ogni mese.

In particolare, le richieste $\mathbf{d_t}$ (espresse in unità di prodotto) vengono organizzate in modo da coprire un orizzonte temporale di 5 mesi, come mostrato nella seguente tabella.

	Mese 1	Mese 2	Mese 3	Mese 4	Mese 5
d_t	270	290	250	240	310

Per la produzione del farmaco si utilizza una linea di produzione a ciclo continuo caratterizzata da costi unitari di produzione mensile $\mathbf{c_t}$ (espresse in migliaia di euro) e da una capacità di produzione mensile $\mathbf{b_t}$ (espresse in unità di prodotto), così come descritto nella seguente tabella.

	Mese 1	Mese 2	Mese 3	Mese 4	Mese 5
c_t	25	28	32	21	24
b_t	250	220	280	270	260

I farmaci eventualmente non consegnati saranno immagazzinati. Ciò comporta un costo aggiuntivo di inventario, il cui valore unitario $\mathbf{h_t}$ (espresso in migliaia di euro) varia mensilmente nel seguente modo:

	Mese 1	Mese 2	Mese 3	Mese 4	Mese 5
h_t	10	12	8	10	11

In magazzino sono disponibili inizialmente 100 unità di prodotto. L'obiettivo dell'azienda è pianificare la produzione del farmaco in modo tale da minimizzare i costi di produzione e di inventario.

Formulare un modello di ottimizzazione che rappresenti e risolva adeguatamente il problema.

APPLICAZIONE SOFTWARE

File → new → OPL project e diamo al progetto il nome "Problema 5".

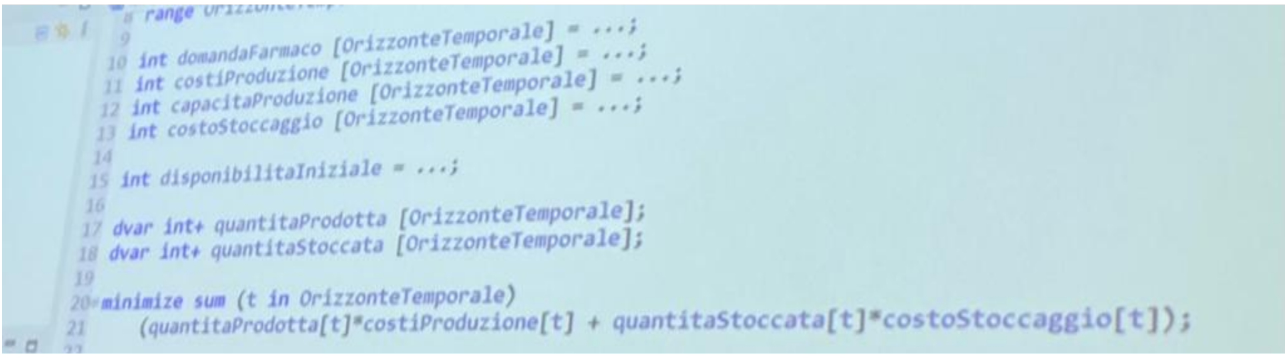
Innanzitutto bisogna dichiarare i dati: abbiamo un orizzonte temporale pari a 5 unità di tempo.

A tal proposito, ci serve una variabile per indicare il numero di intervalli nell'orizzonte di pianificazione che chiamiamo T:

int T = …

Dato che ci serve indicizzare dei vettori in questo orizzonte temporale, dobbiamo prima utilizzare quello che ieri abbiamo chiamato range, che in questo caso va da 1 a T:

range OrizzonteTemporale = 1. . T



```
8  range Orizzonte...
9
10 int domandaFarmaco [OrizzonteTemporale] = ...;
11 int costiProduzione [OrizzonteTemporale] = ...;
12 int capacitaProduzione [OrizzonteTemporale] = ...;
13 int costoStoccaggio [OrizzonteTemporale] = ...;
14
15 int disponibilitaIniziale = ...;
16
17 dvar int+ quantitaProdotta [OrizzonteTemporale];
18 dvar int+ quantitaStoccata [OrizzonteTemporale];
19
20 minimize sum (t in OrizzonteTemporale)
21     (quantitaProdotta[t]*costiProduzione[t] + quantitaStoccata[t]*costoStoccaggio[t]);
```

Nelle righe 10, 11, 12 e 13 sono stati definiti 4 vettori indicizzati in orizzonte temporale, che rappresentano i vettori della traccia: di domanda, di costi produzione, di capacità e di costo stoccaggio.

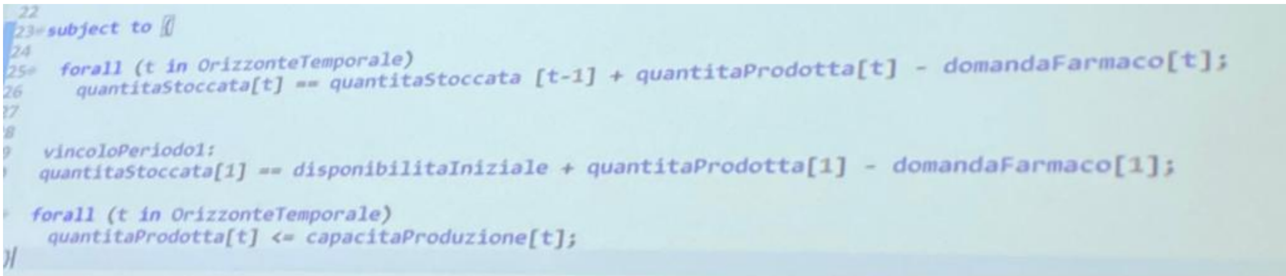
Nella riga 15 viene creata una nuova variabile, definita da un'altra parte.

Variabili decisionali

A questo punto andiamo a definire le variabili decisionali (righe 17 e 18) che ci servono a capire quanto viene prodotto e quanto viene stoccato. Le variabili decisionali in questione sono intere e non negative; inoltre, anche queste variabili sono indicizzate in orizzonte temporale.

Funzione obiettivo

La funzione obiettivo (righe 20 e 21) è la minimizzazione dei costi complessivi di gestione della produzione del magazzino. Quindi, viene minimizzata la somma, fatta sull'orizzonte di pianificazione t in T, della quantità prodotta al tempo t per i relativi costi di produzione al tempo t e la quantità stoccata al tempo t per il relativo costo di stoccaggio al tempo t.



```
22
23 subject to {
24
25   forall (t in OrizzonteTemporale)
26     quantitaStoccata[t] == quantitaStoccata [t-1] + quantitaProdotta[t] - domandaFarmaco[t];
27
28
29   vincoloPeriodo1:
     quantitaStoccata[1] == disponibilitaIniziale + quantitaProdotta[1] - domandaFarmaco[1];

   forall (t in OrizzonteTemporale)
     quantitaProdotta[t] <= capacitaProduzione[t];
}
```

Vincoli

Nei vincoli bisogna indicare che, per ogni istante di tempo t, la quantità stoccata (=quello che si ha nel magazzino) è uguale a quello che si aveva all'inizio dell'istante t (t-1) + quello che viene prodotto nell'istante t - la domanda del farmaco nell'istante t.
NB. Se poniamo T = 0, di conseguenza nel vincolo avremo quantità stoccata di -1, che non esiste in quanto il vettore è indicizzato da 0 a T. Questo è importante per capire come si può condizionare un vincolo.

Dopodiché, al tempo 1 bisogna imporre che la quantità stoccata sia pari alla disponibilità iniziale + la quantità prodotta al tempo 1 - la domanda del farmaco al tempo 1.

[*Nella riga 29 (vincoloPeriodo1:) si può osservare come si etichetta un vincolo: lo si può fare semplicemente mettendo il nome del vincolo e i due punti : prima del vincolo stesso. Questo sarà utile nella prossima lezione*].

Infine, l'ultimo vincolo da aggiungere riguarda che, per ogni istante di tempo t, la quantità prodotta non deve superare la capacità di produzione.

Apriamo Excel e creiamo il file Problema_5.

	A	B	C	D	E	F	G	H	I	J
1										
2	T	1	2	3	4	5		dispIniziale	100	
3										
4	d_t	270	290	250	240	310		T	5	
5										
6	c_t	25	28	32	21	24				
7										
8	b_t	250	220	280	270	260				
9										
10	h_t	10	12	8	10	11				

Dati | Risultati

Con T andiamo a definire i periodi di tempo: 1, 2, 3, 4, 5.

Con d_t, c_t, b_t e h_t riportiamo tutti i dati che troviamo nel testo del problema 5.

Rinominiamo questo foglio "Dati".

Creiamo un secondo foglio "Risultati":

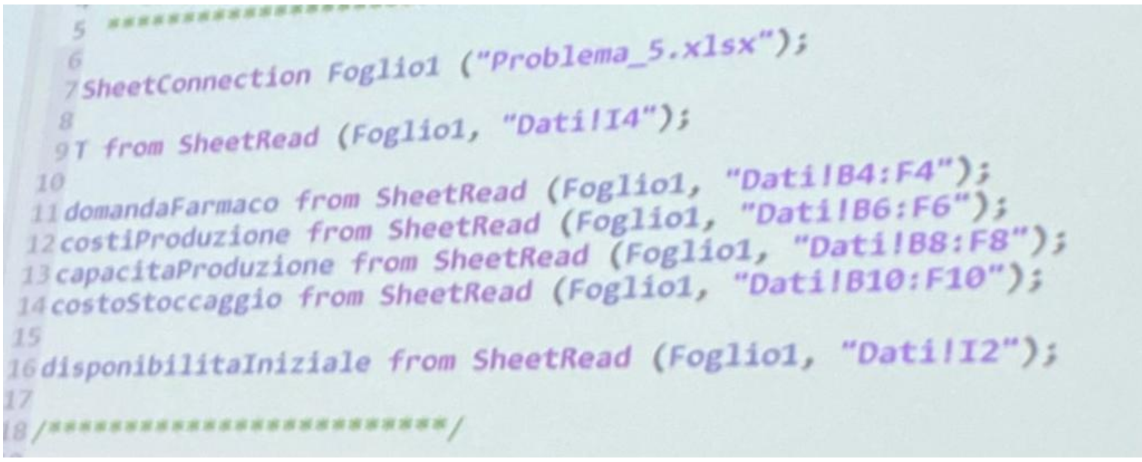
	A	B	C	D	E	F	G
1							
2	T	1	2	3	4	5	
3							
4	Q Prodotta						
5							
6	Q Stoccata						
7							
8							
9	Costo complessivo						

Dati | Risultati

Dopo aver salvato il file e chiuso Excel, torniamo sul software e, dove è presente il progetto “problema 5”, si clicca sul tasto destro e su proprietà in modo incollarvi il file Excel.

Si ritorna sul software, si clicca sul tasto destro e poi su refresh: il progetto comparirà sotto.

A questo punto dobbiamo far sì che il modello prenda i dati dal foglio Excel caricato.



Utilizziamo SheetConnection che è un oggetto di tipo connessione ad un foglio e indica come tipicamente viene chiamato un costruttore: nome della tipologia progetto, nome dello specifico oggetto/dell’istanza che serve a gestire i dati (in questo caso Foglio1).

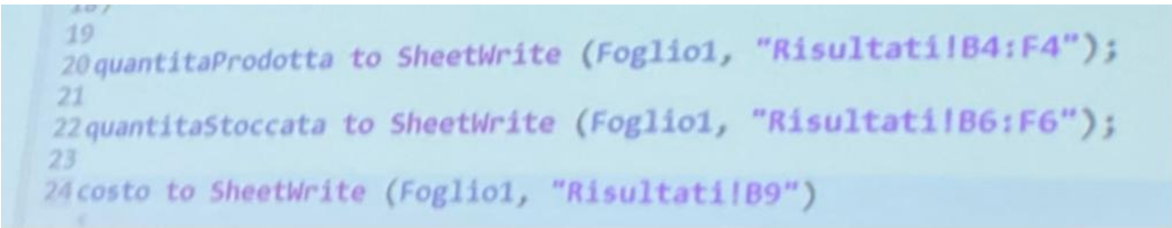
from SheetRead prende in ingresso un collegamento ad un foglio e una posizione.

Nella riga 9 indichiamo che la variabile T deve essere letta dal Foglio1 (in particolare sul foglio “Dati” su Excel) nella posizione I4.

Nella riga 11 indichiamo che la variabile domandaFarmaco deve essere letta da Foglio1 (in particolare sul foglio “Dati” su Excel) nella posizione B4F4.

La stessa cosa viene fatta nelle righe 12, 13, 14 e 16 per le altre variabili.

Adesso andiamo a scrivere sul foglio “Risultati” su Excel utilizzando to SheetWrite.



Le variabili in questione sono quantitaProdotta, quantitaStoccata e costo che verranno scritte su Foglio1 (in particolare sul foglio “Risultati” su Excel) nelle posizioni indicate.

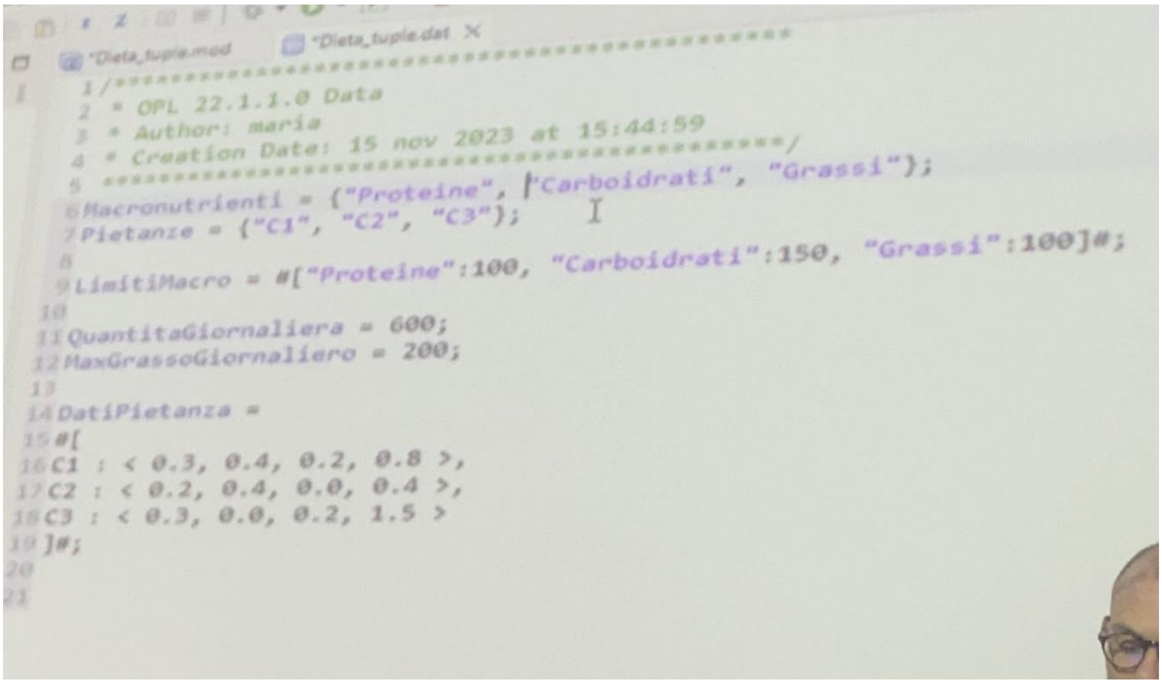
Adesso riapriamo il file Excel e troviamo i risultati nelle rispettive caselle:

	A	B	C	D	E	F
1						
2	T	1	2	3	4	5
3						
4	Q Prodotta	240	220	270	270	260
5						
6	Q Stoccata	70	0	20	50	0
7						
8						
9	Costo complessivo	34070				

Dati | Risultati

SECONDO PROBLEMA: QUANTITA' DI MACRONUTRIENTI NELLE PIETANZE

Nel caso di questo specifico problema, quello che bisogna fare è scegliere l'insieme delle pietanze da far mangiare ad una persona andando a rispettare dei limiti per quanto riguarda le quantità dei macronutrienti.



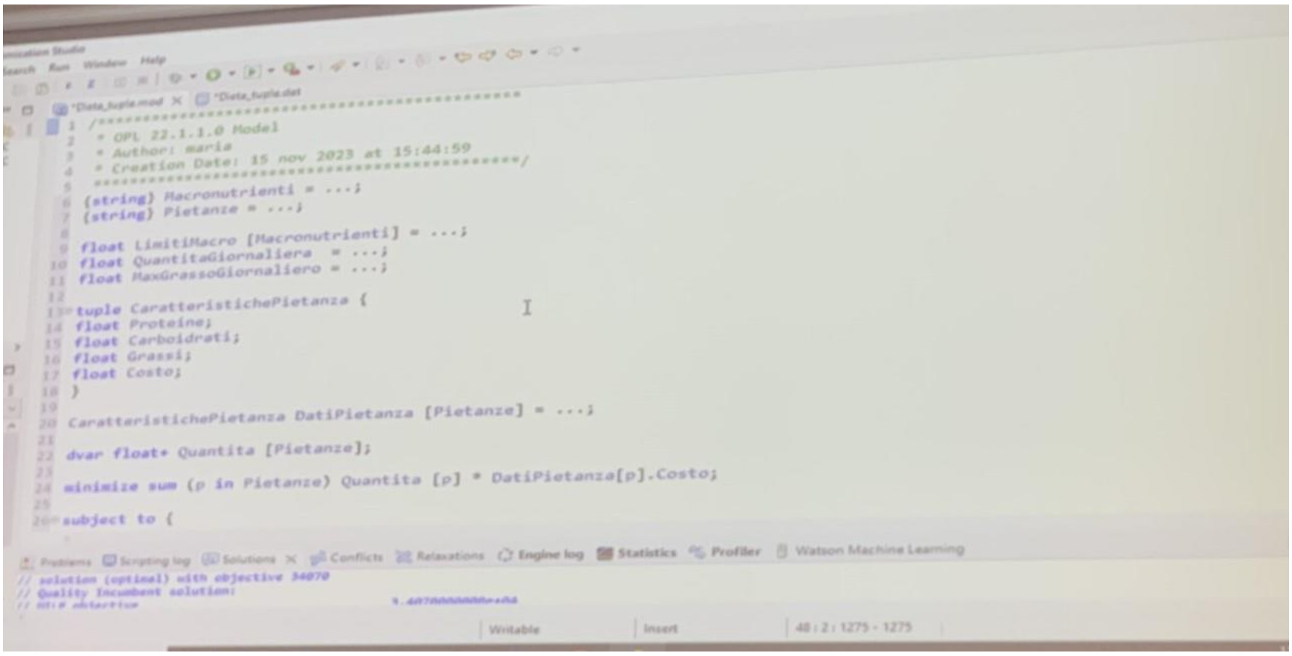
```
1 /*********************************************
2  * OPL 22.1.1.0 Data
3  * Author: maria
4  * Creation Date: 15 nov 2023 at 15:44:59
5  *********************************************/
6 Macronutrienti = {"Proteine", "Carboidrati", "Grassi"};
7 Pietanze = {"C1", "C2", "C3"};
8
9 LimitiMacro = #["Proteine":100, "Carboidrati":150, "Grassi":100]#;
10
11 QuantitaGiornaliera = 600;
12 MaxGrassoGiornaliero = 200;
13
14 DatiPietanza =
15 #[
16 C1 : < 0.3, 0.4, 0.2, 0.8 >,
17 C2 : < 0.2, 0.4, 0.0, 0.4 >,
18 C3 : < 0.3, 0.0, 0.2, 1.5 >
19 ]#;
20
21
```

Nelle righe 6 e 7 della foto soprastante andiamo a definire gli elementi che compongono rispettivamente Macronutrienti (Proteine, Carboidrati e Grassi) e Pietanze (C1, C2, C3).

La riga 9 è un vettore che va ad indicare quelli che sono i limiti di ciascuno dei macronutrienti da inserire nelle pietanze.

Nella riga 11 è indicata la quantità giornaliera di pietanza da ingerire, cioè 600 gr, mentre nella riga 12 è indicata la quantità max di grassi che deve essere presente nella pietanza, cioè 200 gr.

Nelle righe 16, 17 e 18 sono indicate le quantità presenti in 100 gr di ciascuna pietanza rispettivamente di proteine, carboidrati e grassi. L'ultimo valore presente in ciascuna tupla indica invece il prezzo per 100 gr di ciascuna delle tre pietanze.



Riga 6: (string) Macronutrienti = …: si dichiarano un insieme di stringhe, che prendono il nome di Macronutrientri. Queste riceveranno, attraverso un file Excel o un foglio dati, una sequenza di valori, racchiusi tra doppi apici. I macronutrienti specifici per questo problema sono le proteine, i carboidrati e i grassi.

Riga 7 (string) Pietanze = …: stesso discorso della sesta riga, con l'insieme di stringhe che prende il nome di Pietanze. Le pietanze specifiche per questo problema saranno indicate in modo generico con C1, C2 e C3.

Nelle righe 9, 10 e 11 sono utilizzati dei **float**, importanti per rappresentare numeri in virgola mobile a precisione singola. Nello specifico si hanno 3 float:

- **float LimitiMacro:** è un vettore inteso come il valore limite che ciascun macronutriente può assumere. Questo vettore è indicizzato in macronutrienti …(**[Macronutrienti]),** cioè ci sarà un elemento di questo vettore che sarà riferito all'insieme di stringhe dei macronutrienti.
- **float QuantitaGiornaliera:** valore singolo che indica la quantità di prodotto che bisogna mangiare al giorno.
- **float MaxGrassoGiornaliero:** valore singolo che indica la quantità massima giornaliera di grassi che bisogna mangiare.

La riga 13 (**tuple CaratteristichePietanza)** va a creare una tupla in cui si andranno ad inserire degli oggetti, cioè in questo caso 4 valori (i float delle righe 14-15-16-17) che prenderanno il nome di proteine, carboidrati, grassi e costi.

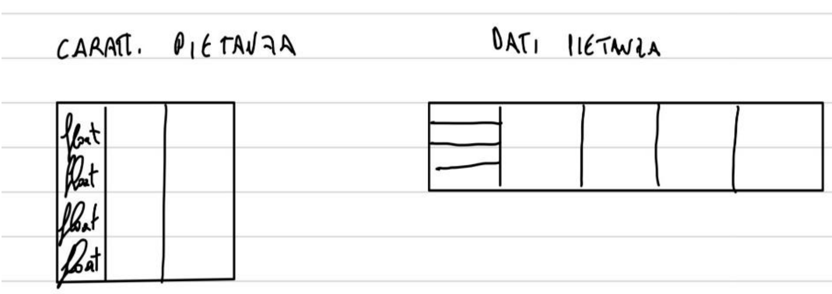
Una tupla all'interno di questo linguaggio di programmazione (Java) è definita come un oggetto che contiene una sequenza di elementi immutabili, che può essere definito dal programmatore. La *tupla può essere immaginata come un contenitore che ha delle caratteristiche definite dallo stesso programmatore, e queste caratteristiche non possono essere cambiate.*

Quindi ora, avendo ottenuto un modello decisionale, si possono creare oggetti di questo tipo utilizzando numeri arbitrari.

Riga 20: CaratteristichePietanza DatiPietanza [Pietanza] = … : anche in questo caso si crea una tupla DatiPietanza, cioè una singola variabile, mentre CaratteristichePietanza indica il nome del tipo di oggetto, cioè un vettore che sarà indicizzato a Pietanze ([Pietanze]).

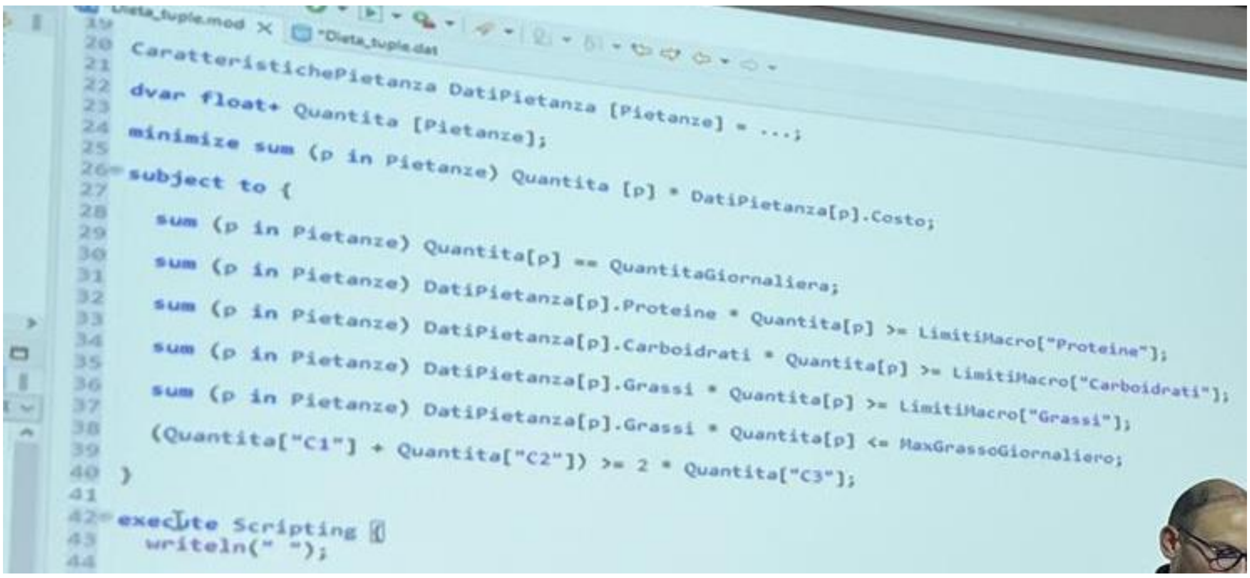
CaratteristichePietanza può essere visto come un contenitore e al suo interno si aspetta di avere quattro variabili float (proteine, carboidrati, grassi e prezzo).

DatiPietanza è un vettore indicizzato in Pietanze, e ognuno degli elementi contenuti in questo vettore fa riferimento a un oggetto contenuto in Pietanze.



Variabili decisionali
A questo punto vengono dichiarate le variabili decisionali. La prima decisione (riga 22) riguarda la quantità di prodotto di ognuna delle pietanze da includere giornalmente nella dieta. Per ogni pietanza inclusa nella dieta viene espressa una decisione. I parametri delle tuple non possono essere utilizzati come variabili decisionali.

Funzione obiettivo
Adesso l'obiettivo è minimizzare il costo delle Pietanze. Ogni pietanza è composta da una quantità di proteine, una di carboidrati, una di grassi e da un costo. All'interno di ogni pietanza si conosce il costo di ciascun macronutriente in quantità di 100 gr.
Per minimizzare il costo si effettua la sommatoria (minimize sum, riga 24) della quantità di quella pietanza (variabile decisionale) e dal costo unitario di ciascuna pietanza (DatiPietanza[p].costo) che è un parametro all'interno del vettore DatiPietanza alla posizione p.



Vincoli

In questo problema decisionale vi sono tre vincoli da considerare: (righe da 26 a 40)

- La quantità che complessivamente si mangia ogni giorno di pietanza deve essere pari alla quantità giornaliera richiesta di quella determinata pietanza.
- Per ognuno dei macronutrienti vi sono dei limiti o dei valori minimi che devono essere contenuti in una pietanza.
  - Nel caso dei carboidrati e delle proteine, essi devono essere maggiori o uguali alla quantità minima richiesta per ogni pietanza.
  - Per quanto riguarda i grassi, vi è invece una quantità massima che deve essere inserita nell'alimentazione, per cui la quantità di grassi dovrà essere minore o uguale della quantità massima richiesta. La quantità di grassi complessiva è data dalla somma delle pietanze per la quantità di prodotto per la percentuale di grassi contenuta.
- La somma della quantità di C1 e C2 ingerita deve essere maggiore o uguale al doppio della quantità di C3 che si mangia.

Si ottengono i seguenti risultati: