ESERCIZIO1

PREMESSA

Per risolvere problemi spesso esistono delle regole che, dai dati del problema, permettono di calcolare o *dedurre* la soluzione. Questa situazione si può descrivere col termine

regola(<sigla>,<lista antecedenti>,<conseguente>)

che indica una regola di nome <sigla> che consente di dedurre <conseguente> conoscendo tutti gli elementi contenuti nella <lista antecedenti>, detta anche *premessa*. Problemi "facili" possono essere risolti con una sola regola; per problemi "difficili" una sola regola non basta a risolverli, ma occorre applicarne diverse in successione.

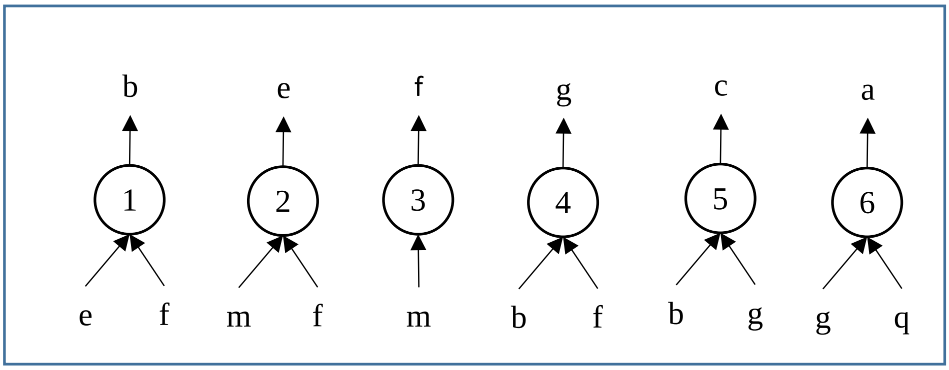
Si considerino le seguenti regole:

regola(1,[e,f],b)	regola(2,[m,f],e)	regola(3,[m],f)
regola(4,[b,f],g)	regola(5,[b,g],c)	regola(6,[g,q],a)

Per esempio la regola 1 dice che si può calcolare (o dedurre) **b** conoscendo **e** ed **f** (cioè gli elementi della lista [e,f]); conoscendo **b** ed **f** (cioè gli elementi della lista [b,f]) è possibile dedurre **g** con la regola 4. Quindi, a partire da **e** ed **f** è possibile dedurre prima **b** (con la regola 1) e poi **g** (con la regola 4).

Un *procedimento di deduzione* (o di calcolo) è rappresentato da un elenco di regole da applicare e quindi può essere descritto dalla lista delle sigle di queste regole. Il procedimento [1,4] descrive la soluzione del problema: "dedurre **g** a partire da **e** ed **f**".

Una maniera grafica per rappresentare le regole è quella mostrata nella seguente figura: consiste nell'associare un albero (rovesciato) ad ogni regola: la radice (in alto) è il conseguente, le foglie (in basso) sono gli antecedenti.



Con questa maniera grafica risolvere il problema "dedurre **g** a partire da **e** ed **f**" è particolarmente facile; si cerca un "albero" (cioè una regola) che ha come radice l'incognita (cioè **g**): in questo caso ne esiste solo uno che è la regola 4: si veda la seguente figura 1 a sinistra.

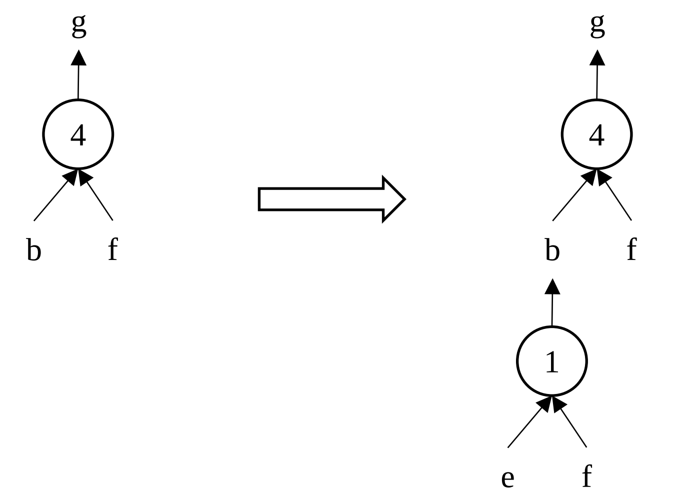


Figura 1

Le foglie di questo albero (**b** ed **f**) *non* sono tutte note: quelle note (**f** in questo caso) sono vere e proprie foglie, quelle incognite (**b** in questo caso) vanno considerati come “anelli” a cui “appendere” un altro albero; quindi bisogna continuare *sviluppando* la foglia incognita **b**, cioè “*appendendo*” a **b** l’albero rappresentato dalla regola 1, come illustrato nella figura 1 a destra.
Adesso tutte le foglie dell’albero così ottenuto (**e** ed **f**) sono note e il problema è risolto. Per costruire la lista occorre *partire dal basso*: prima si applica la regola 1, che utilizza solo i dati; poi si può applicare la regola 4. Il procedimento è quindi [1,4].

Come altro esempio, in figura 2 è illustrata la soluzione del problema: “dedurre **e** a partire da **m**”. Tale soluzione si ottiene costruendo successivamente i due alberi mostrati; il procedimento è [3,2].

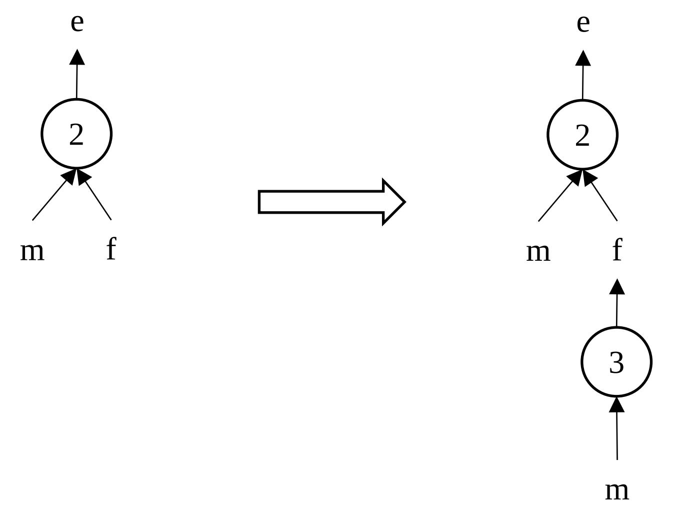


Figura 2

N.B. Nelle liste richieste occorre elencare le sigle delle regole nell’ordine che corrisponde alla sequenza di applicazione: la prima (a sinistra) della lista deve essere la sigla che corrisponde alla prima regola da applicare (che ha come antecedenti solo dati); l’ultima (a destra) deve essere la sigla che ha come conseguente l’elemento incognito da dedurre richiesto dal problema.

In ogni procedimento di deduzione, l'applicazione di una regola rende disponibile il conseguente da utilizzare (come antecedente) per poter applicare regole successive: la prima regola è sempre applicabile a partire *solo* dai dati e non ci sono regole *ripetute*.
Inoltre, ad ogni passo del procedimento, se ci fossero più regole applicabili contemporaneamente, nella lista occorre dare la precedenza a quella con sigla inferiore.

PROBLEMA
Sono date le seguenti regole:

regola(1,[r,u],x)	regola(2,[f,q],g)	regola(3,[p,r],u)
regola(4,[a,f],c)	regola(5,[d],f)	regola(6,[u,x],a)
regola(7,[n,g],p)	regola(8,[p,g],d)	regola(9,[y,w],d)
regola(10,[f,d],a)	regola(11,[f],q)	regola(12,[f,g],n)

Trovare:
1. la lista L1 che descrive il procedimento per dedurre **a** conoscendo **p** e **r**;
2. la lista L2 che descrive il procedimento per dedurre **c** conoscendo **y** e **w**;
3. la lista L3 che descrive il procedimento per dedurre **d** conoscendo **f**.

L1	
L2	
L3	

SOLUZIONE

L1	[3,1,6]
L2	[9,5,10,4]
L3	[11,2,12,7,8]

COMMENTI ALLA SOLUZIONE
Prima domanda. Risulta immediato che **a** è deducibile con la regola 6 (da **u** e **x**) e con la regola 10 (da **f** e **d**). Si può decidere facilmente a favore della prima alternativa perché **u** e **x** sono facilmente deducibili dai dati in due passaggi; la lista che corrisponde al procedimento è [3,1,6]

Seconda domanda. L'incognita **c** è deducibile solo con la regola 4, che ha come antecedenti **a** e **f**; questa volta per **a** è conveniente usare la regola 10 perché questa ha come antecedente **f**, che deve essere dedotto comunque, e **d**. Quest'ultimo è deducibile dai dati con la regola 9 e **f** è deducibile da **d** con la regola 5; la lista che corrisponde al procedimento è [9,5,10,4].

Terza domanda. L'incognita **d** è deducibile con due regole: la 8 e la 9; è facile escludere quest'ultima perché un antecedente, **y**, non è un dato e non è deducibile con alcuna regola. Degli antecedenti della regola 8, **p** e **g**, il primo è deducibile (solo) con la regola 7, che richiede **n** e **g**; **n**, a sua volta è deducibile (solo) con la regola 12, che richiede **f** (dato) e ancora **g**; questo è deducibile (solo) con la regola 2, che richiede **f** (dato) e **q**. Con la regola 11 si chiude il procedimento deduttivo, la cui lista è quindi [11,2,12,7,8].

N.B. **d** è deducibile da **f**, come si è visto, inoltre **f** è deducibile da **d** (regola 5): si dice che **f** e **d** sono *logicamente equivalenti*; la conoscenza di uno "equivale" alla conoscenza dell'altro.

ESERCIZIO 2

PREMESSA
In un deposito di minerali esistono esemplari di vario peso e valore individuati da sigle di riconoscimento. Ciascun minerale è descritto da una termine che contiene le seguenti informazioni:

tab(<sigla del minerale>, <valore in euro>, <peso in Kg>).

Il deposito contiene i seguenti minerali:

tab(m1,50,54) tab(m2,48,57) tab(m3,55,33)
tab(m4,48,57) tab(m5,45,52) tab(m6,49,59)
tab(m7,47,52) tab(m8,52,54) tab(m9,47,59)

PROBLEMA
Disponendo di un autocarro con portata massima di 85 Kg, trovare la lista L1 delle sigle di 2 minerali diversi trasportabili con questo autocarro che consente di trasportare il massimo valore possibile.
Disponendo di un autocarro con portata massima di 140 Kg, trovare la lista L2 delle sigle di 3 minerali diversi trasportabili con questo autocarro che consente di trasportare il massimo valore possibile.
Disponendo di un autocarro con portata massima di 195 Kg, trovare la lista L3 delle sigle di 4 minerali diversi trasportabili con questo autocarro che consente di trasportare il massimo valore possibile.

N.B. Nelle liste, elencare le sigle in ordine crescente; per le sigle si ha il seguente ordine: m1<m2<… <m9.

L1	
L2	
L3	

SOLUZIONE

L1	[m3,m7]
L2	[m3,m7,m8]
L3	[m1,m3,m7,m8]

COMMENTI ALLA SOLUZIONE
In generale, un metodo per risolvere il problema (detto della *forza bruta*) è quello di generare tutte le combinazioni di 2, di 3 e di 4 minerali presi tra i nove del deposito, calcolarne peso e valore e scegliere, tra quelle trasportabili, quella che ha valore maggiore; poiché tali combinazioni sono rispettivamente (9×8)/(2×1) = 36, (9×8×7)/(3×2×1) = 84, (9×8×7×6)/(4×3×2×1) = 126, tale metodo è pesante (cioè richiede molti "calcoli" e molto "spazio").

Per particolari problemi esistono comunque modi "più veloci", detti *euristici* che consentono di (costruire ed) esaminare un minor numero di combinazioni.
In questo problema sono "poche" le combinazioni trasportabili: quindi basta mettere i minerali in ordine (crescente) rispetto al peso.

minerale	valore	peso
m3	55	33
m5	45	52
m7	47	52
m1	50	54
m8	52	54
m4	48	57
m2	48	57
m6	49	59
m9	47	59

e disporli in maniera crescente di peso (che non è la maniera richiesta di scrivere le liste!).

Per la prima domanda, si vede che ci sono solo (le “prime”) 2 coppie trasportabili
[m3,m5] 33+52=85
[m3,m7] 33+52=85

Per la seconda domanda, ci sono solo (le “prime”) 5 terne trasportabili:
[m3,m5,m7] 33+52+52=137
[m3,m5,m1] 33+52+54=139
[m3,m5,m8] 33+52+54=139
[m3,m7,m1] 33+52+54=139
[m3,m7,m8] 33+52+54=139

Per la terza domanda, ci sono solo (le “prime”) 6 quaterne trasportabili:
[m3,m5,m7,m1] 33+52+52+54=191
[m3,m5,m7,m8] 33+52+52+54=191
[m3,m5,m7,m4] 33+52+52+57=194
[m3,m5,m7,m2] 33+52+52+57=194
[m3,m5,m1,m8] 33+52+54+54=193
[m3,m7,m1,m8] 33+52+54+54=193

ESERCIZIO 3

PROBLEMA
You are given 16 numbers; the first 7 numbers have an average of 6.00; 6 numbers (distinct from the before mentioned ones) have an average of 7.00; the last 3 numbers have an average of 20.00. What is the average of all 16 numbers?
Put your answer in the box below as a decimal number with two decimal places; use a dot as decimal mark.

SOLUZIONE
$\boxed{9.00}$

COMMENTI ALLA SOLUZIONE
La somma dei primi 7 numeri è $7\times6.00 = 42.00$; la somma degli altri 6 numeri è $6\times7.00 = 42.00$; la somma degli ultimi 3 numeri è $3\times20.00 = 60.00$; la somma totale è $42.00+42.00+60.00 = 144.00$.
Ciò implica che la media dei 16 numeri sia $144.00/16 = 9.00$.

ESERCIZIO 4

PROBLEMA
Si consideri la seguente procedura PROVA1.

```
procedure PROVA1;
variables A, B, K, J integer;
A ← 0;
B ← 0;
for J from 1 to 4 step 1 do
      for K from 1 to 5 step 1 do
            A ← A+ J;
            B ← B + K;
      endfor;
endfor;
output A,B;
endprocedure;
```

Determinare il valore di output di A e B.

A	
B	

SOLUZIONE

A	50
B	60

COMMENTI ALLA SOLUZIONE
È facile verificare che, all'uscita del ciclo "for" interno i valori di J, A e B sono dati dalla seguente tabella.

J	A	B
1	5	15
2	15	30
3	30	45
4	50	60

All'uscita del ciclo interno, B aumenta di valore ogni volta di una quantità pari alla somma dei primi 5 numeri naturali (15); A, invece, aumenta di valore di una quantità che è cinque volte il valore di J.

ESERCIZIO 5

PREMESSA

La ripetizione di un gruppo di azioni può essere comandata non solo con la struttura “for” già vista, ma anche con la struttura “while”, illustrata dal seguente esempio.

```
B ← 10;
A ← 0;
K ← 0;
while A < B do
    K ← K + 1;
    A ← K × K + A;
endwhile;
output A;
```

Se il predicato A < B è vero, il ciclo viene ripetuto; quando diventa falso si passa alla esecuzione della istruzione successiva a “endwhile”. In questo caso il valore di B rimane fisso a 10, mentre quello di A cambia dopo ogni iterazione assumendo i seguenti valori: 1, 5, 14. Dopo la terza iterazione il valore di A non è più minore di quello di B e il ciclo si arresta: in output si ha quindi 14.

PROBLEMA

Si consideri la seguente procedura PROVA2.

```
procedure PROVA2;
variables A, B, K, Z integer;
A ← 1;
B ← 0;
K ← 0;
input Z;
while A > B do
    K ← K +1;
    A ← A + Z × K;
    B ← K × K + B;
endwhile;
output A,B;
endprocedure;
```

Determinare il valore di output di A e B, ordinatamente per tre valori di input per Z: 3, 6, 9.

Valore in input per Z	Valore in output per A	Valore in output per B
3		
6		
9		

SOLUZIONE

Valore in input per Z	Valore in output per A	Valore in output per B
3	46	55
6	271	285
9	946	1015

COMMENTI ALLA SOLUZIONE
Quando Z vale 3, il corpo del ciclo “while” viene ripetuto 5 volte e le variabili K, A, B assumono i valori riportati nella tabella seguente.

K	A	B
1	4	1
2	10	5
3	19	14
4	31	30
5	46	55

In generale (per ogni valore di Z) il valore di K “conta” l’*iterazione* (cioè conta quante volte è stato eseguito il ciclo “while”), il valore di B aumenta del valore di K al quadrato e il valore di A aumenta del valore di Z moltiplicato il valore di K.

ESERCIZIO 6

PROBLEMA
Si consideri la seguente procedura PROVA3.

```
procedure PROVA3;
variables A, B, K, Z integer;
Z ← 0;
for K from 1 to 3 step 1 do
        A ← 40;
        B ← 0;
        Z ← Z + 2;
        while A > B do
                A ← A + Z × K;
                B ← (Z + K) × (Z + K) + B;
        endwhile;
        output A, B;
endfor;
endprocedure;
```

Determinare le tre coppie di valori di output per A e B.

	A	B
prima coppia di valori		
seconda coppia di valori		
terza coppia di valori		

SOLUZIONE

	A	B
prima coppia di valori	52	54
seconda coppia di valori	56	72
terza coppia di valori	58	81

COMMENTI ALLA SOLUZIONE
Il ciclo “while” viene ripetuto 6 volte quando K vale 1, 2 volte quando K vale 2 e una sola volta quando K vale 3; i valori delle variabili, prima di “endwhile”, sono mostrati nella seguente tabella.

K	Z	A	B
1	2	42	9
1	2	44	18
1	2	46	27
1	2	48	36
1	2	50	45
1	2	52	54
2	4	48	36
2	4	56	72
3	6	58	81