ESERCIZIO 1

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,f],c)

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 deduttivo, o di calcolo) è rappresentato da un *insieme di regole da applicare in sequenza opportuna* per dedurre un certo elemento (incognito) a partire da certi dati: 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 1: consiste nell’associare un albero (rovesciato) ad ogni regola: la radice (in alto) è il conseguente, le foglie (in basso) sono gli antecedenti.

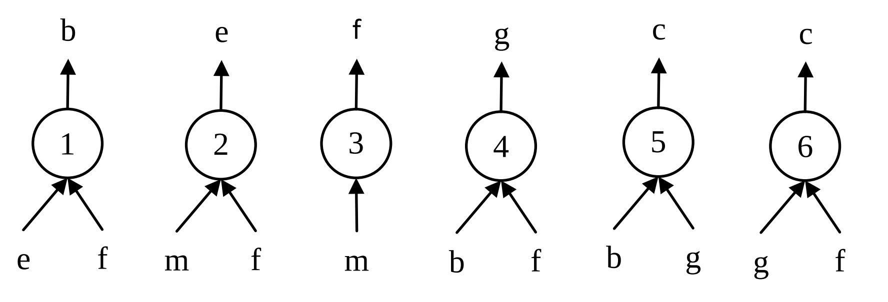

Figura 1

Con questa rappresentazione 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 figura 2 a sinistra.

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 2 a destra.

Adesso tutte le foglie dell'albero così ottenuto (**e** ed **f**) sono note e il problema è risolto.

Si può anche dire che un albero le cui foglie sono tutte note rappresenta un procedimento per dedurre la "radice" a partire dalle "foglie". Per costruire la lista corrispondente 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 (individuato dalla lista) [1,4].

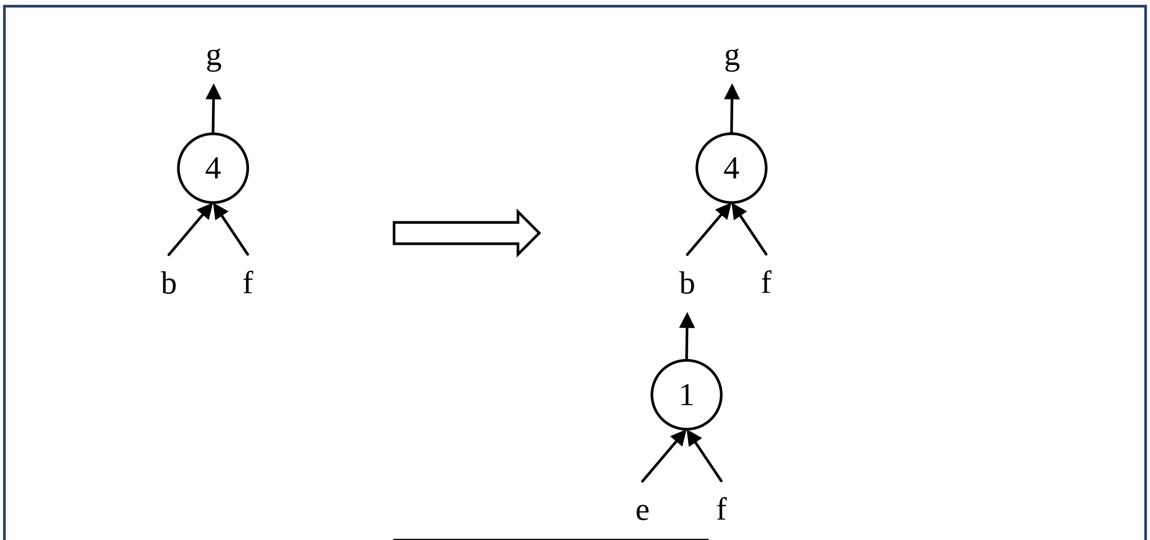

Figura 2

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 della regola che ha come conseguente l'elemento incognito da dedurre.

Nella lista non ci sono regole *ripetute* (infatti un procedimento di deduzione è un *insieme* di regole da applicare in opportuna sequenza). L'applicazione di una regola rende disponibile il conseguente da utilizzare (come antecedente) nell'applicazione di regole successive.

La lista associata a un (ben preciso) procedimento si costruisce quindi per passi successivi a partire dal primo elemento che è la sigla della prima regola da applicare; ad ogni passo, se ci fossero più regole applicabili, occorre dare la precedenza (nella lista) a quella con sigla *inferiore* (questo per rendere *unica* la lista associata al procedimento).

N.B. In alcuni casi esistono più procedimenti deduttivi possibili che permettono di ricavare un certo elemento dagli stessi dati, in maniere diverse (cioè con alberi diversi e quindi con insiemi diversi di regole). Per esempio il problema "dedurre **c** a partire da **b** ed **f**" (dalle regole viste sopra) ha due distinti procedimenti risolutivi; gli alberi relativi ai due procedimenti sono mostrati nella seguente figura 3.

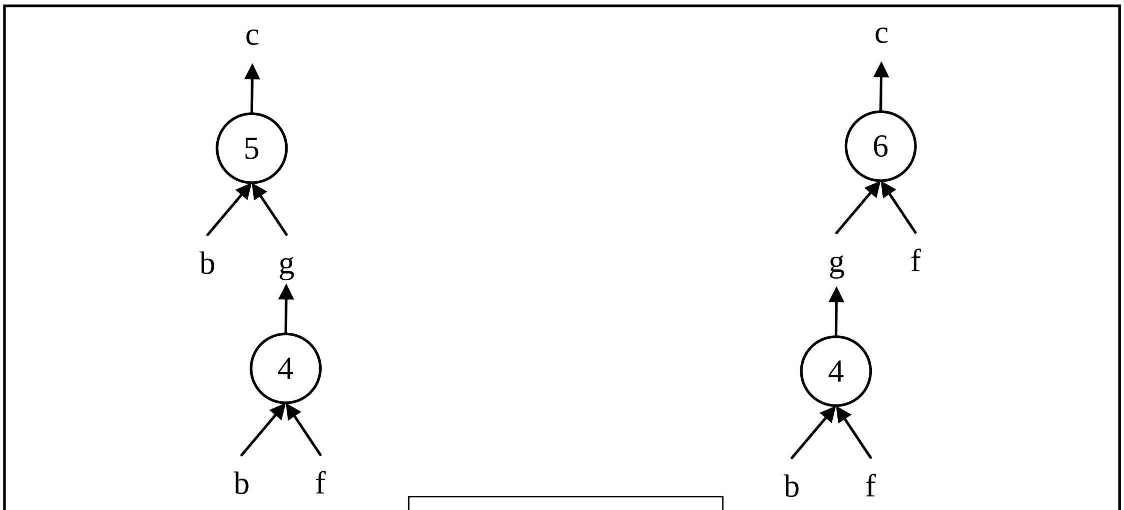

Figura 3

Le liste associate sono, rispettivamente, [4,5] e [4,6].

In un procedimento deduttivo, il numero di regole *differenti* coinvolte (e, quindi, anche il numero di elementi della lista corrispondente al procedimento) si dice *lunghezza* del procedimento.

PROBLEMA
Sono date le seguenti regole:

regola(1,[m,n,u],v)	regola(2,[u,v],z)	regola(3,[v,w],z)
regola(4,[t],u)	regola(5,[p,q],u)	regola(6,[m],u)
regola(7,[u,w],z)	regola(8,[t,u],v)	regola(9,[r],v)
regola(10,[p,t],r)	regola(11,[p,q,u],w)	regola(12,[n,o],m)

Trovare:
1. la lista L1 che descrive il procedimento per dedurre **z** conoscendo **p** e **q**;
2. la lista L2 che descrive il procedimento per dedurre **z** conoscendo **n** e **o**;
3. il numero N di modi diversi per dedurre **z** conoscendo **t** e **w**.

L1	[ ]
L2	[ ]
N	

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 un termine che contiene le seguenti informazioni:

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

Il deposito contiene i seguenti minerali:

tab(m1,55,84)	tab(m2,53,86)	tab(m3,58,87)
tab(m4,56,83)	tab(m5,52,82)	tab(m6,54,88)
tab(m7,57,81)	tab(m8,51,88)	tab(m9,59,89)

PROBLEMA
Disponendo di un autocarro con portata massima di 165 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 170 Kg, trovare la lista L2 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 255 Kg, trovare la lista L3 delle sigle di 3 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	[ ]

ESERCIZIO 3

PROBLEMA
John has a lock with three dials; each dials has the symbols A, B, C, D, E, F, G and H. John is a very smart boy, but he is constantly forgetting the combination of his lock. If he can try one lock combination every two seconds, how long will it take to him to try every possible lock combination?
Put your answer in the fields below, keeping in mind that each field is a two digits field: for example 2 hours, 0 minutes and 12 seconds should be written as 02:00:12.

	:		:	

ESERCIZIO 4

PROBLEMA
Si consideri la seguente procedura PROVA1.

```
procedure PROVA1;
variables A, K integer;
A ← -1;
K ← 0;
while A < 0 do
        K ← K + 1;
        A ← 2 × K × K × K – 40 × K × K + K + 1;
endwhile;
output K, A;
endprocedure;
```

Determinare il valore di output di K e A.

K	
A	

ESERCIZIO 5

PROBLEMA
Si consideri la seguente procedura PROVA2.

```
procedure PROVA2;
variables A, B, K integer;
A ← 0;
B ← 100;
K ← 0;
while A<B do
        K ← K +1;
        A ← A + K^2;
        B ← B – K + 1;
endwhile;
output A, B, K;
endprocedure;
```

Determinare i valori di output.

N.B. Il simbolo ^ denota l’elevamento a potenza.

A	
B	
K	

ESERCIZIO 6

PROBLEMA
Si consideri la seguente procedura PROVA3.

```
procedure PROVA3;
variables A, P, Q, R, K integer;
input A;
P ← A;
Q ← A;
R ← A;
for K from 1 to 9 step 1 do
    input A;
    if A > P then Q ← P; P ← A;
        else if A > Q then Q ← A; endif;
    endif;
    if A<R  then R ← A;  endif;
endfor;
output P, Q, R;
endprocedure;
```

Se i valori di input per A sono 12, 9, 15, 3, 9, 3, 1, 8, 12, 13 calcolare i valori di output.

P	
Q	
R	