

**Scuola secondaria di secondo grado - GARA 5 APR '12**

**ESERCIZIO 1 (SISTEMI DI REGOLE) G5-2012-SEC-SG-01**

**PREMESSA**

Con il termine

$regola(\langle sigla \rangle, \langle lista\ antecedenti \rangle, \langle conseguente \rangle, \langle peso \rangle)$

si può descrivere una *regola* (di deduzione) che consente di dedurre il *conseguente* conoscendo tutti gli elementi contenuti nella *lista degli antecedenti*; ogni regola è poi identificata in modo univoco da una sigla e ha un *peso*, che dà l'idea di quanto sia oneroso applicarla. Per esempio, dato il seguente insieme di regole:

regola(1,[c1,c2],i,12)	regola(2,[i,h],a,3)	regola(3,[h,p1],c1,2)
regola(4,[h,p2],c2,7)	regola(5,[c1,c2],a,4)	regola(6,[p1,p2],h,3)
regola(7,[p1,p2],i,2)	regola(8,[c1,i],c2,8)	regola(9,[i,a],h,6)

si osserva che, conoscendo gli elementi contenuti nella lista  $[p1,p2]$ , è possibile dedurre (direttamente) **h** con la regola 6 e **i** con la regola 7; ma conoscendo  $[p1,p2]$  è anche possibile dedurre **c1** applicando prima la regola 6 (per dedurre **h**) e poi la regola 3 (conoscendo ora  $[h,p1]$ ). Si può quindi dire che la lista  $[6,3]$  rappresenta un *procedimento* per dedurre **c1** da  $[p1,p2]$ ; la lista contiene infatti l'indicazione delle regole che devono essere applicate. Per esempio, la lista  $[6,3,4,5]$  rappresenta un procedimento per calcolare **a** da  $[p1,p2]$ . Sommando i pesi delle regole applicate è possibile ottenere una *valutazione* del procedimento; pertanto, si può affermare che il procedimento  $[6,3,4,5]$  per dedurre **a** da  $[p1,p2]$  ha valutazione 16. La lista  $[1,X,Y]$  descrive il procedimento per calcolare **h** conoscendo i valori di **c1** e **c2**, se si sostituisce 5 a X e 9 a Y.

Naturalmente in un procedimento *tutti* i dati *parziali* (quelli via via dedotti) devono essere utilizzati e devono essere prodotti una sola volta (cioè non si devono fare "passi" inutili).

N.B. Nella lista che descrive il procedimento, la prima sigla indica la prima regola che deve essere applicata.

**PROBLEMA**

È dato il seguente insieme di regole (in cui il nome del termine è "rs" invece di "regola"):

rs(1,[e,f,c],d,3)	rs(2,[b,c],a,5)	rs(3,[c,g],b,7)	rs(4,[f,g],c,2)
rs(5,[g,h],e,7)	rs(6,[q],h,4)	rs(7,[p],f,1)	rs(8,[m,g],f,6)
rs(9,[c,f],p,7)	rs(10,[p],a,4)	rs(11,[c,f],q,1)	rs(12,[m,r],c,6)
rs(13,[g],m,7)	rs(14,[x,r],g,9)	rs(15,[g,m],p,5)	rs(16,[m],g,7)

Dato **m**, si deve derivare **a**:

- trovare la lista L1 che descrive il procedimento la cui valutazione K1 è minima,
- trovare la lista L2 che descrive il procedimento la cui valutazione K2 è massima.

L1	
L2	
K1	
K2	

**SOLUZIONE**

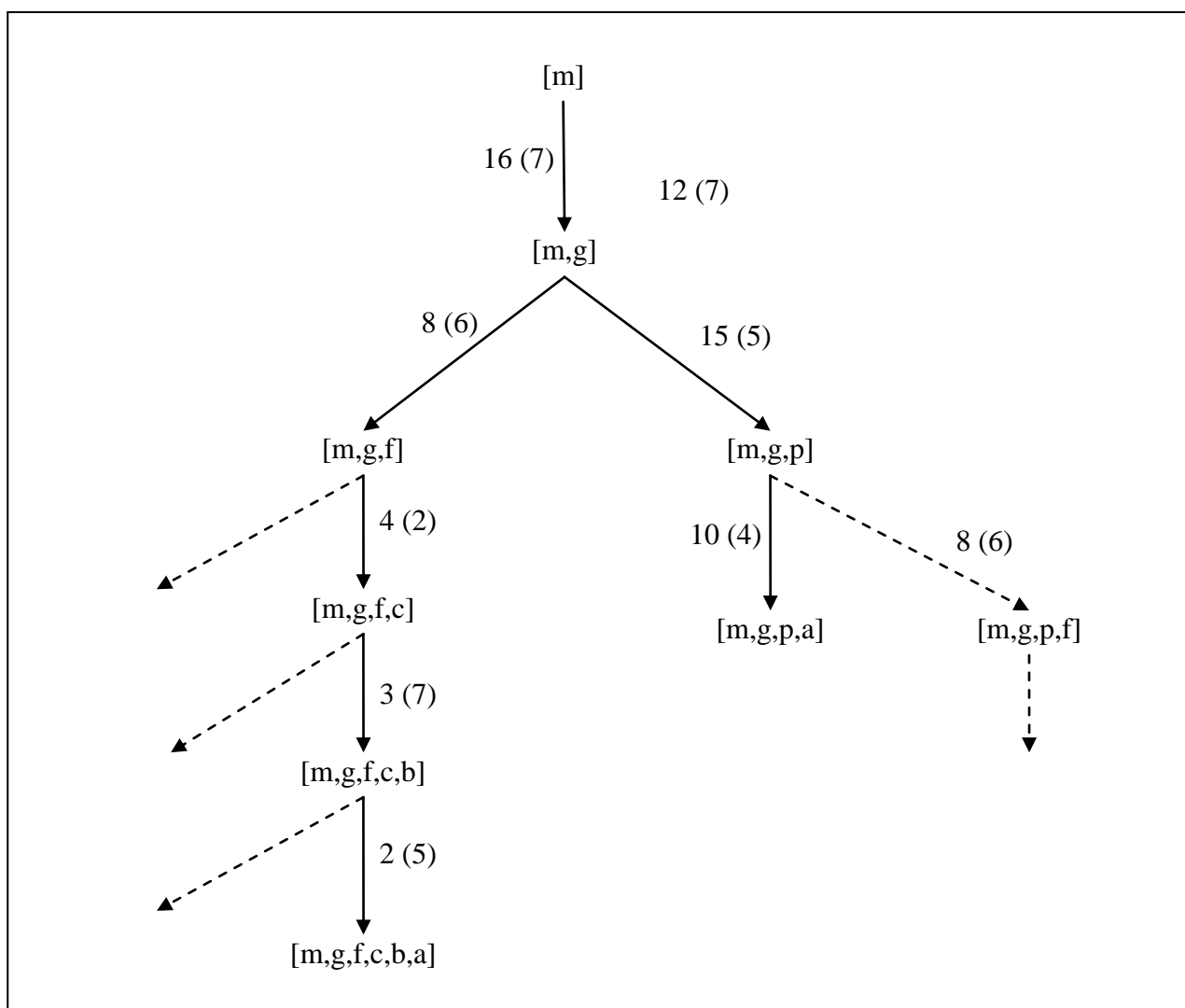
L1	[16,15,10]
L2	[16,8,4,3,2]
K1	16
K2	27

Scuola secondaria di secondo grado - GARA 5 APR '12

COMMENTI ALLA SOLUZIONE

Si osservi la figura seguente che rappresenta un albero con i nodi costituiti dagli elementi conosciuti (dati o via via dedotti) e i rami costituiti dalle regole che si possono applicare con quelle conoscenze; partendo dalla radice (il dato [m]) si può applicare solo la regola: 16 ottenendo [m,g]. Per ottenere **a** occorre poter applicare la regola 2, che richiede i dati [b,c], o la regola 10 che richiede [p]; bisogna quindi sviluppare l'albero in modo che i rami terminali siano, appunto, 2 o 10.

N.B. La figura *non* è completa: per poter dedurre che i procedimenti mostrati hanno valutazione minima e massima, si sarebbero dovuti mostrare *tutti* i percorsi che terminano con una lista contenente **a**, hanno rami (regole) non ripetuti e tutti i dati parziali utilizzati.



N.B. La figura (non completa!) descrive un metodo che si dice *forward* (o *bottom up*) e consiste nel partire dai dati e usare le regole applicabili per aumentare la conoscenza via via fino a comprendere l'incognita; viene tipicamente impiegato nel "contesto della giustificazione": quando cioè si voglia esporre (o dimostrare) ad altri (o a se stessi) un risultato. In realtà, per trovare un procedimento di soluzione in generale si utilizza anche il metodo *backward* (o *top down*) che consiste nel partire dalla incognita e cercare di individuare una regola per derivarla. Se esiste una regola le cui premesse sono tutte note (i dati), il problema si risolve con una regola (vedi primo esempio descritto nella

**Scuola secondaria di secondo grado - GARA 5 APR '12**

premessa); altrimenti la ricerca continua per trovare (tutte) le regole che consentono di derivare l'antecedente o gli antecedenti non noti (vedi secondo esempio nella premessa).

Scuola secondaria di secondo grado - GARA 5 APR '12

ESERCIZIO 2 (PSEUDOPROGRAMMI) G5-2012-SEC-SG-02

PROBLEMA

Per descrivere una procedura di calcolo viene spesso usato un pseudolinguaggio che utilizza parole inglesi e simboli matematici. Compresa la sequenza dei calcoli descritti nell'esempio che segue, eseguire le operazioni indicate utilizzando i dati di input sotto riportati e trovare il valore delle variabili di output.

```
procedure PROVA;  
variables P, N, I, Z integer;  
input N;  
Z ← 0;  
P ← 1;  
for I from 1 to N do  
    P ← I×P;  
    Z ← Z+P+I×(I+1);  
endfor;  
output P, Z;  
endprocedure;
```

Calcolare i 3 valori di P e i 3 di Z corrispondenti ai valori 3, 4 e 5 di N riempiendo la tabella seguente.

N	P	Z
3		
4		
5		

SOLUZIONE

N	P	Z
3	6	29
4	24	73
5	120	223

COMMENTI ALLA SOLUZIONE

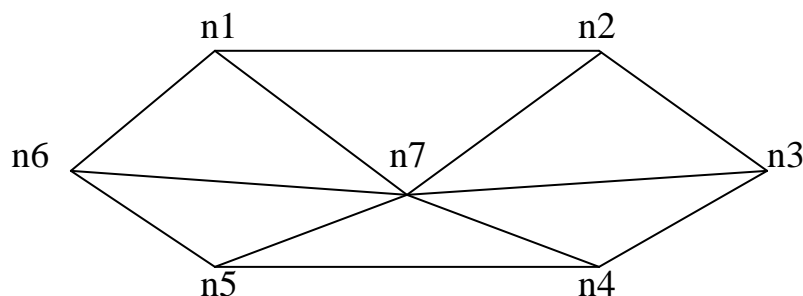
È sufficiente considerare il caso in cui N vale 5; il ciclo all'interno del **for** è ripetuto 5 volte e la tabella seguente riporta i valori di I, P e Z alla fine di ogni ripetizione.

I	P	Z
1	1	3
2	2	11
3	6	29
4	24	73
5	120	223

**ESERCIZIO 3 (GRAFI) G5-2012-SEC-SG-03**

**PREMESSA**

Il seguente grafo stradale



può essere descritto dal seguente insieme di termini (ciascuno dei quali definisce un arco tra due nodi del grafo con la indicazione della relativa distanza)

- |              |              |              |              |              |              |
|--------------|--------------|--------------|--------------|--------------|--------------|
| $a(n1,n2,2)$ | $a(n2,n3,5)$ | $a(n3,n4,3)$ | $a(n4,n5,4)$ | $a(n5,n6,2)$ | $a(n6,n1,3)$ |
| $a(n1,n7,8)$ | $a(n2,n7,6)$ | $a(n3,n7,1)$ | $a(n4,n7,9)$ | $a(n5,n7,7)$ | $a(n6,n7,4)$ |

Un percorso tra due nodi del grafo può essere descritto con la lista dei nodi che lo compongono ordinati dal nodo di partenza al nodo di arrivo. Per esempio, la lista  $[n5,n7,n2,n1]$  descrive un percorso dal nodo  $n5$  al nodo  $n1$  di lunghezza  $K = 15$ .

**PROBLEMA**

Disegnare il grafo stradale corrispondente al seguente insieme di termini (che hanno nome “as” invece di “a”):

- |               |               |                |                |                 |
|---------------|---------------|----------------|----------------|-----------------|
| $as(n8,n2,2)$ | $as(n2,n3,6)$ | $as(n2,n4,3)$  | $as(n9,n3,5)$  | $as(n7,n2,5)$ . |
| $as(n3,n7,7)$ | $as(n4,n7,9)$ | $as(n8,n5,10)$ | $as(n1,n9,4)$  | $as(n6,n1,7)$ . |
| $as(n5,n4,6)$ | $as(n5,n6,3)$ | $as(n6,n7,1)$  | $as(n5,n7,15)$ |                 |

Trovare le liste  $L1$  e  $L2$ , dei percorsi rispettivamente più breve e più lungo fra il nodo  $n1$  e il nodo  $n9$  che passano per 6 nodi intermedi.

N.B. Percorsi senza cicli: per ogni nodo del grafo si può passare (al più) una sola volta.

<b>L1</b>	
<b>L2</b>	

**SOLUZIONE**

<b>L1</b>	$[n1, n6, n5, n4, n2, n7, n3, n9]$
<b>L2</b>	$[n1, n6, n5, n7, n4, n2, n3, n9]$

**COMMENTI ALLA SOLUZIONE**

Dopo qualche tentativo si può disegnare il grafo, come nella figura seguente.

I cammini da  $n1$  a  $n9$  di 8 nodi (quindi con 6 nodi intermedi) e le loro lunghezze sono i seguenti:

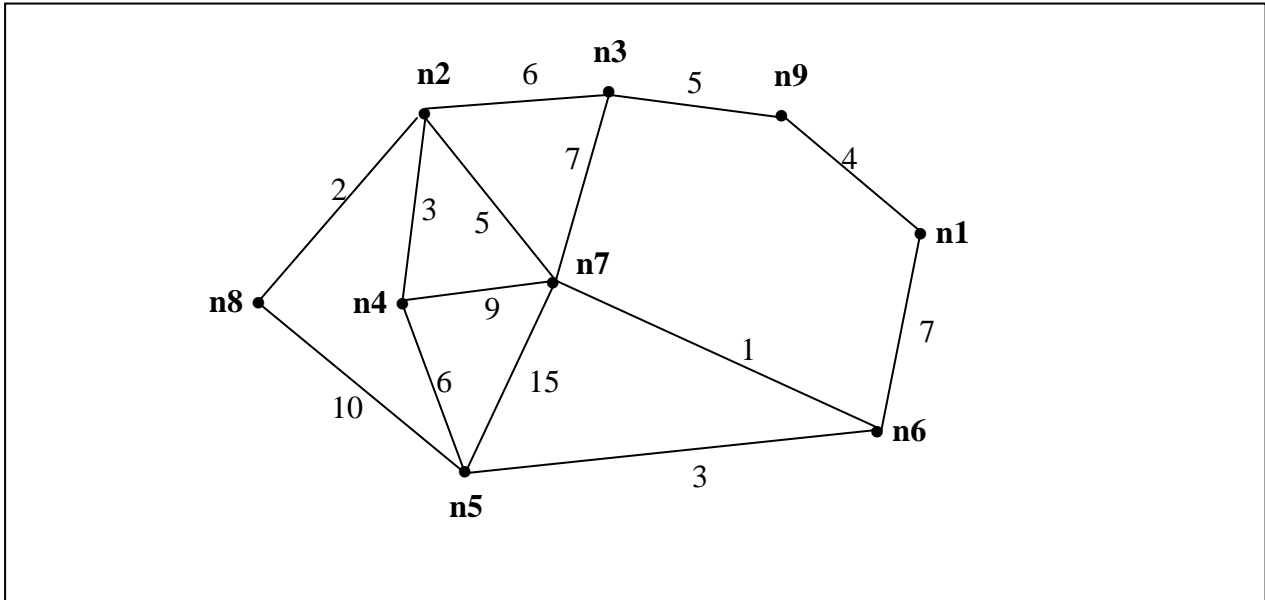
- $[n1, n6, n5, n4, n2, n7, n3, n9]$ , 36;
- $[n1, n6, n5, n8, n2, n7, n3, n9]$ , 39;
- $[n1, n6, n5, n7, n4, n2, n3, n9]$ , 48;
- $[n1, n6, n7, n5, n4, n2, n3, n9]$ , 43;

Scuola secondaria di secondo grado - GARA 5 APR '12

[n1, n6, n7, n5, n8, n2, n3, n9], 46;

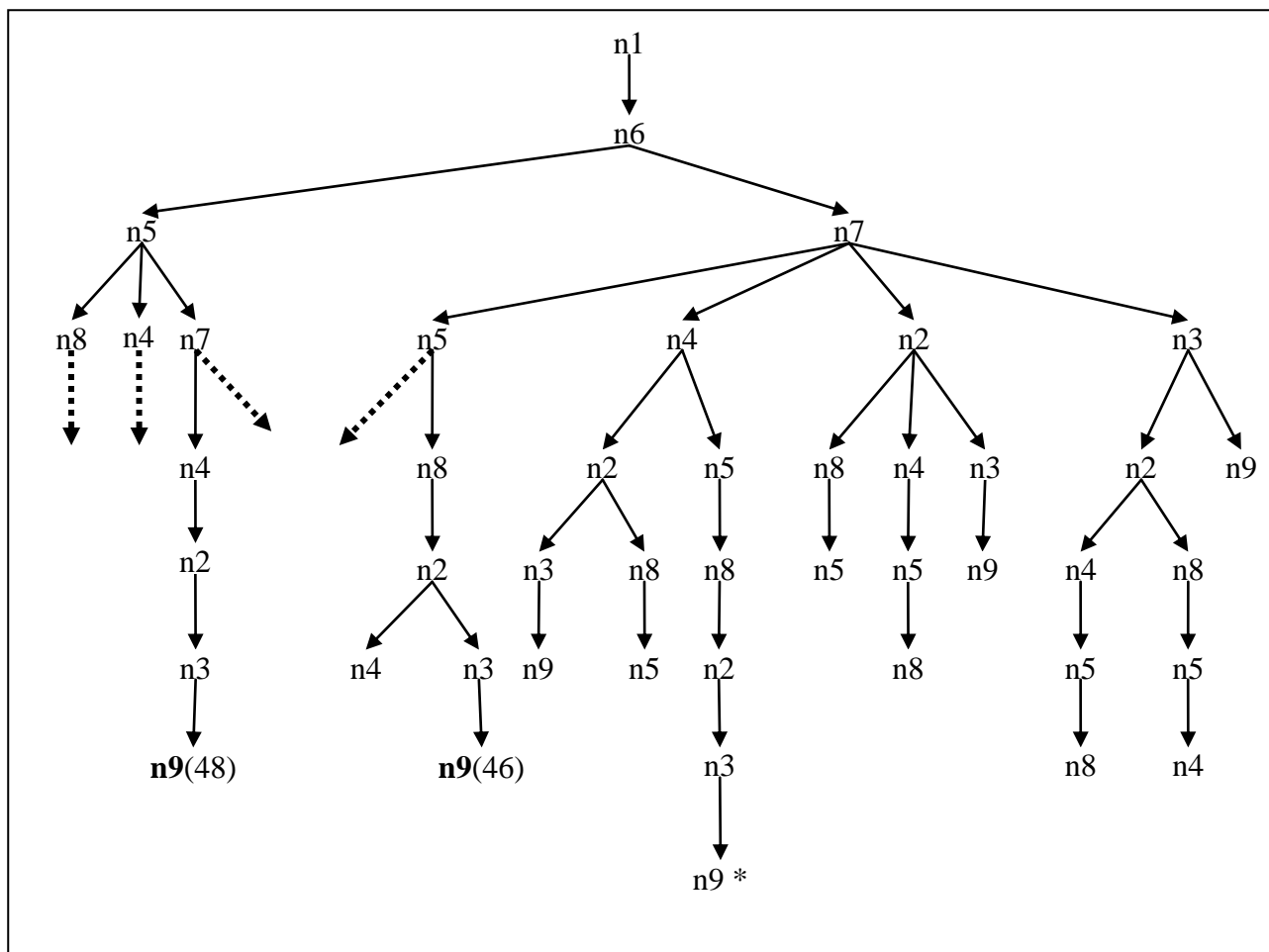
[n1, n6, n5, n4, n7, n2, n3, n9], 41.

Quindi il primo e il terzo sono le due soluzioni.



Come si può essere sicuri di avere elencati tutti i cammini?

Si osservi la figura seguente.



In questa figura è rappresentato (parzialmente!) un *albero di percorsi*; il punto di inizio del percorso è la radice dell'albero; a un nodo **n** dell'albero si connettono come figli i nodi che sono connessi a lui nel grafo, a meno che compaiono già nel percorso dalla radice ad **n**. Il ramo termina quando è raggiunto il nodo finale, oppure quando tutti i nodi connessi con la foglia, nel grafo, compaiono nel percorso dalla radice alla foglia. Il numero massimo di nodi in ogni ramo è il numero di nodi del grafo. In figura il percorso che termina con la foglia **n9 \*** non è una soluzione del problema perché tocca 7 nodi intermedi; sono candidati i percorsi individuati da foglie **n9** (in grassetto) che portano anche la lunghezza.

Per rispondere correttamente al problema occorre costruire *completamente* l'albero dei percorsi con radice n1, per individuare i percorsi con le caratteristiche richieste. Esistono algoritmi che non costruiscono (e non conservano) tutti i rami, ma solo quelli che sono "promettenti" come soluzione.

**Scuola secondaria di secondo grado - GARA 5 APR '12**

**ESERCIZIO 4 (DATA BASE) G5-2012-SEC-SG-04**

**PREMESSA**

Per gestire gli articoli in vendita presso un grande magazzino vengono utilizzate quattro tabelle il cui contenuto è descritto dai quattro termini seguenti:

- tab1(<sigla dell'articolo>,<disponibilità all'apertura>,<prezzo di vendita>)
- tab2(<sigla dell'articolo>,<sigla del fornitore>,<prezzo di acquisto>)
- tab3(<sigla dell'articolo>,<tipo merceologico>,<reparto>)
- tab4(<sigla dell'articolo>,<disponibilità alla chiusura>)

A fine giornata, la situazione di queste tabelle è la seguente:

tab1(a72,140,40)	tab1(a11,120,32)	tab1(a23,220,44)	tab1(a24,130,40)
tab1(a13,190,20)	tab1(a26,180,50)	tab1(a27,145,45)	tab1(a21,110,35)
tab1(a29,210,60)	tab1(a22,240,70)	tab1(a31,140,65)	tab1(a32,220,75)
tab1(a15,145,50)	tab1(a34,120,40)	tab1(a35,210,60)	tab1(a25,220,60)
tab2(a72,g1,10)	tab2(a11,g1,15)	tab2(a23,g3,20)	tab2(a24,g3,30)
tab2(a13,g3,5)	tab2(a26,g3,30)	tab2(a27,g3,40)	tab2(a21,g1,25)
tab2(a29,g2,30)	tab2(a22,g2,60)	tab2(a31,g2,45)	tab2(a32,g1,35)
tab2(a15,g3,20)	tab2(a34,g2,25)	tab2(a35,g2,15)	tab2(a25,g3,30)
tab3(a72,a,x)	tab3(a11,a,y)	tab3(a23,b,y)	tab3(a24,b,x)
tab3(a13,a,z)	tab3(a26,c,y)	tab3(a27,a,y)	tab3(a21,a,y)
tab3(a29,b,x)	tab3(a22,c,y)	tab3(a31,b,x)	tab3(a32,c,z)
tab3(a15,a,z)	tab3(a34,b,x)	tab3(a35,a,z)	tab3(a25,b,y)
tab4(a72,40)	tab4(a11,50)	tab4(a23,80)	tab4(a24,60)
tab4(a13,60)	tab4(a26,50)	tab4(a27,40)	tab4(a21,20)
tab4(a29,140)	tab4(a22,100)	tab4(a31,30)	tab4(a32,40)
tab4(a15,20)	tab4(a34,50)	tab4(a35,50)	tab4(a25,100)

Da queste tabelle si ricavano per esempio le seguenti informazioni: l'articolo a25 appartiene al tipo merceologico **b**, proviene dal fornitore g3, ne sono stati venduti 120 esemplari con un guadagno unitario di 30 euro e guadagno giornaliero di 3600 euro.

**PROBLEMA**

Trovare:

- la lista L1 degli articoli distribuiti dal fornitore g1,
- la lista L2 degli articoli esposti nel reparto **x**,
- la sigla X1 dell'articolo distribuito dal fornitore g3 che realizza il minor guadagno per unità di prodotto,
- la sigla X2 dell'articolo esposto nel reparto **y** che realizza il maggior incasso giornaliero,
- la lista L3 degli articoli che realizzano un guadagno per unità di prodotto maggiore di 25.

NB. Gli elementi di una lista vanno riportati in ordine alfabetico crescente; per esempio:

a21<a22<a23,...; f01<f02<f03<f04<...;

quando una lista non contiene elementi, si dice che la lista è *vuota* e si scrive [] (parentesi quadra aperta seguita *immediatamente* da parentesi quadra chiusa).



**Scuola secondaria di secondo grado - GARA 5 APR '12**

L1	
L2	
X1	
X2	
L3	

**SOLUZIONE**

L1	[a11,a21,a32,a72]
L2	[a24,a29,a31,a34,a72]
X1	a27
X2	a25
L3	[a15,a25,a29,a32,a35,a72]

**COMMENTI ALLA SOLUZIONE**

Lista L1 degli articoli distribuiti dal fornitore g1: [a72, a11, a21, a32].

Lista L2 degli articoli esposti nel reparto x: [a72, a24, a29, a31, a34].

Articoli del fornitore g3 e relativi guadagni per unità di prodotto:

a23, 24  
a24, 10  
a13, 15  
a26, 20  
a27, 5: articolo col guadagno minore  
a15, 30  
a25, 30

Articoli esposti nel reparto y e relativi incassi giornalieri:

a11, 1190  
a23, 3360  
a26, 2600  
a27, 525  
a21, 900  
a22, 1400  
a25, 3600: articolo con l'incasso giornaliero maggiore

Lista L3 degli articoli che realizzano un guadagno per unità di prodotto maggiore di 25: [a72, a29, a32, a15, a35, a25].

**Scuola secondaria di secondo grado - GARA 5 APR '12**

**ESERCIZIO 5 (CACCIA AL TESORO) G5-2012-SEC-SG-05**

**PREMESSA**

Un campo di gara per robot ha la forma di un foglio a quadretti o celle; le celle possono contenere ostacoli che impediscono al robot di attraversarle, oppure dei premi; una cella contiene un tesoro.

					■	2		🏆
		■					■	
		9	1	■		■		4
		👤	7					■

Con riferimento alla figura, il robot (indicato con una sagoma umana) si trova nella cella individuata dalle coordinate (3,2), terza colonna da sinistra e seconda riga dal basso. Il tesoro, rappresentato da una coppa, è nella cella (9,5); il campo contiene 6 ostacoli, individuati da un quadrato nero. I premi sono descritti da 3 numeri: i primi due individuano la cella e il terzo rappresenta il bonus; in questo esempio i premi sono i seguenti: (4,2,7), (3,3,9), (4,3,1), (9,3,4), (7,5,2). Il robot può spostarsi di una cella verso destra o verso l'alto, cioè ad ogni passo solo una delle sue coordinate può aumentare di una unità. In questo esempio, il robot può raggiungere il tesoro (solo) attraverso 4 percorsi L1, L2, L3, L4 individuati dalla lista delle coordinate delle caselle attraversate:

- 1) L1 = [(3,2),(3,3),(4,3),(4,4),(5,4),(6,4),(7,4),(7,5),(8,5),(9,5)], premi raccolti 12,
- 2) L2 = [(3,2),(4,2),(4,3),(4,4),(5,4),(6,4),(7,4),(7,5),(8,5),(9,5)], premi raccolti 10,
- 3) L3 = [(3,2),(4,2),(5,2),(6,2),(6,3),(6,4),(7,4),(7,5),(8,5),(9,5)], premi raccolti 9,
- 4) L4 = [(3,2),(4,2),(5,2),(6,2),(7,2),(8,2),(8,3),(9,3),(9,4),(9,5)], premi raccolti 11.

Per decretare il migliore, ad ogni percorso viene assegnato un punteggio dato dalla somma dei premi raccogliibili su quel percorso; la graduatoria dei percorsi è quindi la seguente: L1, L4, L2, L3.

**PROBLEMA**

La partenza è nella cella (1,1) e il tesoro si trova nella cella (8,8);

la lista dei premi è la seguente:

[(1,2,10),(1,7,10),(2,1,20),(4,6,8),(4,7,4),(5,5,16),(6,4,26),(6,7,6),(7,5,40),(7,6,50)];

la lista degli ostacoli è la seguente:

[(2,4),(3,1),(3,3),(3,6),(4,4),(4,8),(5,2),(5,7),(6,3),(6,5),(7,7),(8,4)].

Trovare il numero N dei percorsi possibili, il punteggio massimo PM e il punteggio minimo Pm.

N	
PM	
Pm	

**SOLUZIONE**

N	12
PM	136
Pm	24

**COMMENTI ALLA SOLUZIONE**

**Scuola secondaria di secondo grado - GARA 5 APR '12**

È facile scoprire, guardando la figura seguente, che ci sono solamente 12 percorsi, i cui punteggi sono: [24, 68, 32, 76, 32, 76, 126, 76, 42, 86, 136, 86].

			■				🏆
10			4	■	6	■	
		■	8			50	
				16	■	40	
	■		■		26		■
		■			■		
10				■			
🏃	20	■					

**Scuola secondaria di secondo grado - GARA 5 APR '12**

**ESERCIZIO 6 (PSEUDOPROGRAMMI) G5-2012-SEC-SG-06**

**PROBLEMA**

Per descrivere una procedura di calcolo viene spesso usato un pseudolinguaggio che utilizza parole inglesi e simboli matematici. Compresa la sequenza dei calcoli descritti nell'esempio che segue, eseguire le operazioni indicate utilizzando i dati di input sotto riportati e trovare il valore delle variabili di output.

```

procedure PROVA;
variables A(1:3), B(1:3), S1, S2, S3, Z, I integer;
input A(1), A(2), A(3), B(1), B(2), B(3);
S1 ← 0;
S2 ← 0;
S3 ← 0;
for I from 1 to 3 do
    S1 ← S1+A(I);
    Z ← A(I)×B(I);
    if Z>0 then S2 ← S2+Z;
    if Z<0 then S3 ← S3+Z;
endfor;
output S1,S2,S3;
endprocedure;
    
```

Calcolare i valori di S1, S2, S3 corrispondenti ai seguenti valori di input

A(1)	A(2)	A(3)	B(1)	B(2)	B(3)
3	-4	5	1	-2	-6

S1	S2	S3

**SOLUZIONE**

S1	S2	S3
4	11	-30

**COMMENTI ALLA SOLUZIONE**

Il ciclo all'interno del **for** viene eseguito tre volte; i valori delle variabili I, S1, S2, S3, Z alla fine di ogni esecuzione sono riportati nella seguente tabella.

I	S1	S2	S3	Z
1	3	3	0	3
2	-1	11	0	8
3	4	11	-30	-30

**Scuola secondaria di secondo grado - GARA 5 APR '12**

**ESERCIZIO 7 (KNAPSACK) G5-2012-SEC-SG-07**

**PROBLEMA**

Nelle lezioni di educazione alimentare, i ragazzi hanno classificato alcuni alimenti in relazione al tipo merceologico, al contenuto proteico e al loro costo. I risultati di questa classificazione sono descritti da una tabella avente la dichiarazione

tabs1(<sigla dell'alimento>, <tipo>, <contenuto proteico>, <costo>).

La tabella, che riporta i dati relativi a un certo numero di alimenti, è la seguente:

tabs1(m11,z,96,145)	tabs1(m22,z,76,144)	tabs1(m2,y,80,131)
tabs1(m20,x,74,130)	tabs1(m5,z,86,150)	tabs1(m14,y,99,150)
tabs1(m7,y,82,138)	tabs1(m8,x,97,151)	tabs1(m9,y,98,149)
tabs1(m10,z,92,140)	tabs1(m21,x,78,159)	tabs1(m12,z,79,130)

Si devono costruire diete con due alimenti di tipo merceologico **z**; trovare:

- la lista L1 delle sigle che corrispondono alla dieta col massimo contenuto proteico;
- la lista L2 delle sigle che corrispondono alla dieta col costo minimo;
- la lista L3 delle sigle che corrispondono alla dieta col massimo contenuto proteico e costo minore di 280.

N.B. Le sigle nelle liste devono comparire in ordine alfabetico *crescente*: m1 prima di m2; m2 prima di m3, ecc.

L1	
L2	
L3	

**SOLUZIONE**

L1	[m10,m11]
L2	[m10,m12]
L3	[m11,m12]

**COMMENTI ALLA SOLUZIONE**

4 ?- ps([z],2,150,400).

Le liste (*non ordinate*) di due alimenti di tipo **z**, con il relativo contenuto proteico e costo (totali) sono:

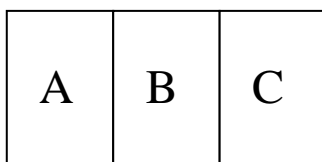
[m22, m11], 172, 289	
[m5, m22], 162, 294	
[m5, m11], 182, 295	
[m10, m5], 178, 290	
[m10, m22], 168, 284	
[m10, m11], 188, 285	contenuto proteico massimo
[m12, m10], 171, 270	costo minimo
[m12, m5], 165, 280	
[m12, m22], 155, 274	
[m12, m11], 175, 275	costo minore di 280 e contenuto proteico massimo

**Scuola secondaria di secondo grado - GARA 5 APR '12**

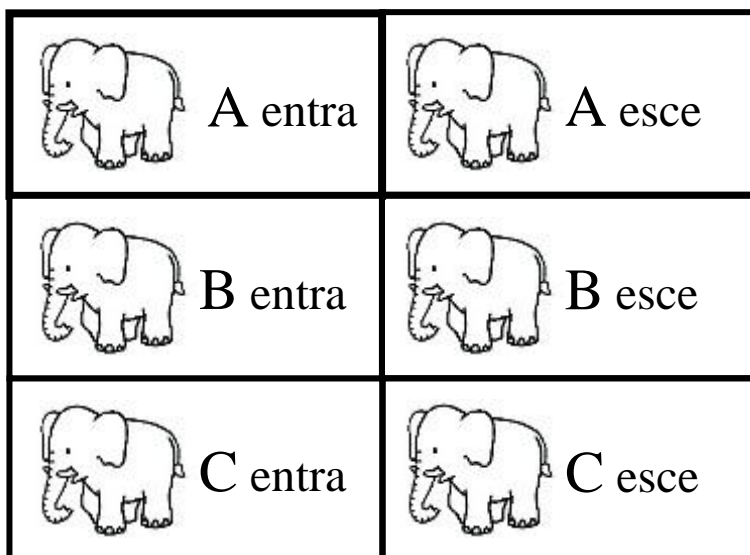
**ESERCIZIO 8 (COMBINATORIA) G5-2012-SEC-SG-08**

**PROBLEMA**

Un impresario di circo ha tre elefanti A, B, C che vuole esibire in uno spettacolo mostrandoli al pubblico in tutte le *combinazioni* possibili (uno per volta, a coppie e tutti insieme). Ogni elefante è addestrato ad occupare una precisa posizione sul palco, come in figura



e ad entrare sul palco e uscirne per mezzo di differenti segnali sonori che possono essere emessi premendo uno dei 6 pulsanti come in figura.



L'impresario vuole che le configurazioni prodotte dagli elefanti si susseguono in ordine alfabetico, partendo e finendo col palco vuoto. Quale è il numero N minimo di volte che occorre premere un pulsante per ottenere ciò?

N	
---	--

**SOLUZIONE**

N	10
---	----

**COMMENTI ALLA SOLUZIONE**

La procedura si può schematizzare con la seguente tabella:

PALCO	PULSANTE/I (da premere per ottenere la configurazione successiva)
-------	---

**Scuola secondaria di secondo grado - GARA 5 APR '12**

vuoto	A entra
A	B entra
AB	C entra
ABC	B esce
AC	A esce, C esce, B entra
B	C entra
BC	B esce
C	C esce
vuoto	

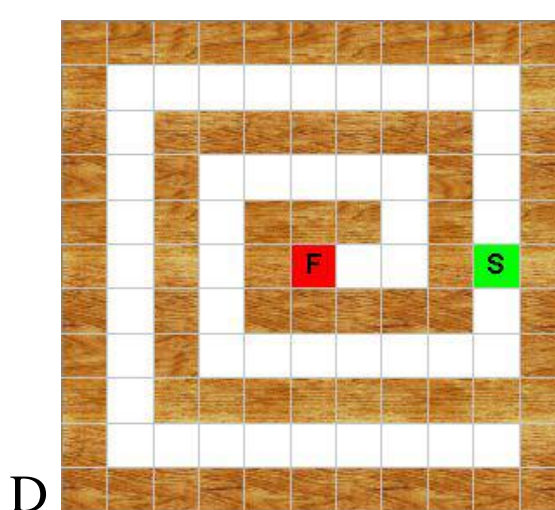
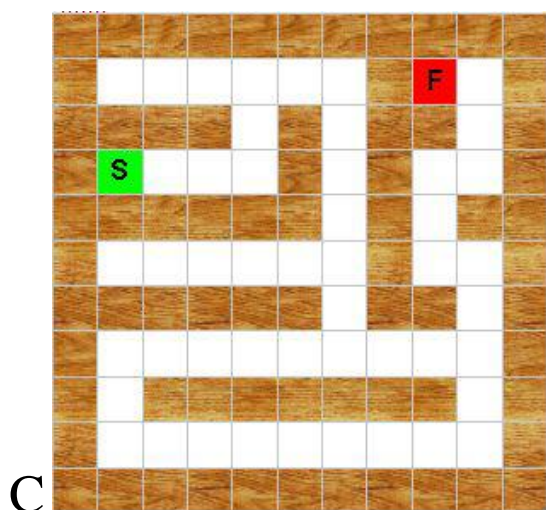
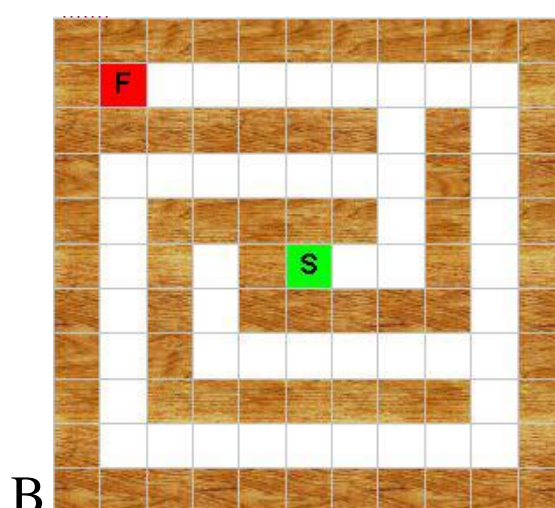
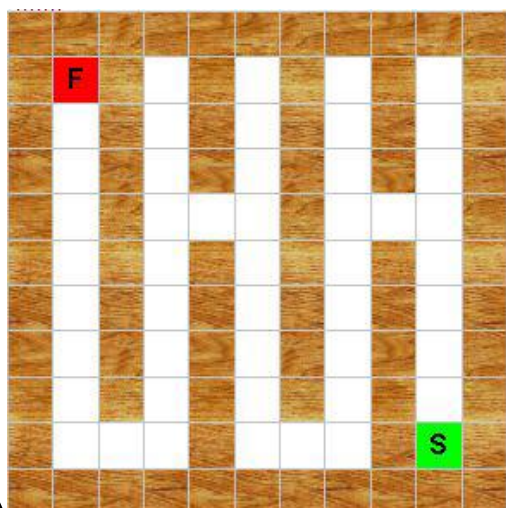
ESERCIZIO 9 (VARIE) G5-2012-SEC-SG-09

PROBLEMA

Un labirinto è, essenzialmente, un percorso intricato (cioè con molti bivi) delimitato da “muri”. È comparso su Internet un algoritmo per “uscire da un labirinto” o meglio per passare da una certa posizione iniziale (marcata con S per *start*) a una certa posizione finale (marcata con F per *final*) muovendosi per *passi discreti*, cioè mediante movimenti di data lunghezza in varie direzioni. L'algoritmo è il seguente.

0. Si comincia nella posizione di partenza, marcata con S.
1. Ci si ferma se si è arrivati alla posizione marcata con F.
2. Se non c'è un muro alla destra, allora girarsi di 90° a destra fare un passo in avanti; ritornare al punto 1 dell'algoritmo.
3. Se possibile, fare un passo avanti; ritornare al punto 1 dell'algoritmo.
4. Girarsi di 90° a sinistra; ritornare al punto 1 dell'algoritmo.

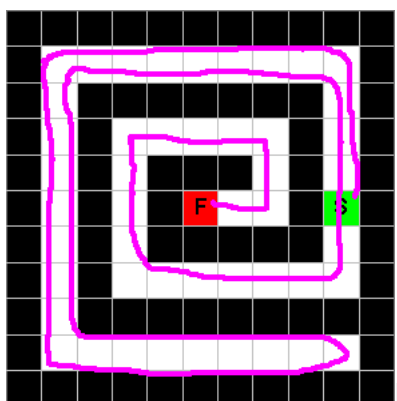
Purtroppo l'algoritmo non è adatto a tutti i labirinti. Quale dei seguenti labirinti non è risolto dall'algoritmo (cioè: seguendo le istruzioni, dalla posizione iniziale non si raggiunge la finale)?



Indicare la risposta (A, B, C, D) nella seguente casella.







ESERCIZIO 10 (COMBINATORIA) G5-2012-SEC-SG-10

PROBLEMA

Mario deve descrivere la soffitta di una casa, che ha 9 stanze rettangolari, e disegna la seguente pianta.

		N		
	0110	0011	0101	
W	1000	1110	0011	E
	0000	1100	1001	
		S		

La stanza in basso a destra della pianta ha una scala interna che conduce al piano inferiore; nessuna delle stanze ha aperture (porte o finestre) verso l'esterno.

In ogni stanza Mario ha aggiunto un codice di quattro cifre binarie per descrivere le porte di ogni stanza; la prima cifra da sinistra si riferisce al nord, la seconda ad est, la terza al sud, la quarta ad ovest. Se la cifra è 0 (oppure 1) vuol dire che nella stanza non c'è (oppure c'è) una porta in quella direzione. Per esempio da una stanza col codice 0100 è possibile andare nella stanza vicina ad est e non ci sono porte verso nord, ovest, sud.

Nella pianta compilata da Mario ci sono degli errori di *incompatibilità* nella disposizione *reciproca* dei codici. Quanti?

N.B. È possibile la risposta 0.

Errori	
--------	--

SOLUZIONE

Errori	2
--------	---

COMMENTI ALLA SOLUZIONE

In una soffitta, naturalmente, non ci possono essere *porte* verso l'esterno (come esplicitamente specificato); se in una stanza c'è una porta verso un'altra, anche da quest'ultima ci deve essere una porta verso la prima.

Nella pianta ci sono due errori entrambi relativi alla stanza a nordest:

1. il codice 0101 dice che c'è una porta che va verso est, impossibile perché verso l'esterno;
2. il codice 0101 dice che c'è una porta che va verso ovest, ma nella stanza a ovest non c'è una porta verso est.

**Scuola secondaria di secondo grado - GARA 5 APR '12**

Il codice 0000 della stanza a sudovest è una stranezza (vuol dire che la stanza non ha porte e quindi è inaccessibile) ma non è *incompatibile* con i codici delle stanze vicine.

**Scuola secondaria di secondo grado - GARA 5 APR '12**

**ESERCIZIO 11 (CASTORO) G5-2012-SEC-SG-11**

**PROBLEMA**

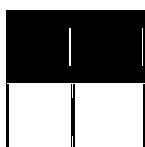
Beaver John has received a secret message (in the form of black and white squares). Unfortunately a part of the message has been destroyed by a spill of red colour.

John is glad, that this case was foreseen and that it was agreed a remedy to that beforehand. There are additional squares in the message. Each square in the rightmost column (column number 6) or the lowest row (row number 6) is black only if the number of preceding black squares in its row, respectively in its column, is odd.

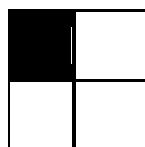
	1	2	3	4	5	6
1						
2						
3						
4						
5						
6						

What is the exact pattern of the red piece?

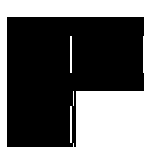
A



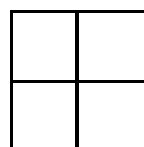
B



C



D



Enter your choice in the box below.

**SOLUZIONE**

**COMMENTI ALLA SOLUZIONE**

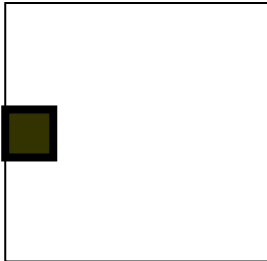
Easy: it's sufficient to check the sums in the sixth row and column.

ESERCIZIO 12 (CASTORO) G5-2012-SEC-SG-12

PREMESSA

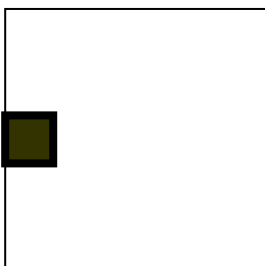
The Beaver loves flowers. May be this is the reason why he has invented a simple programming language for visual design based on the idea of plant life.

Each picture starts with a square called **a**; for example the bold square in the following box:

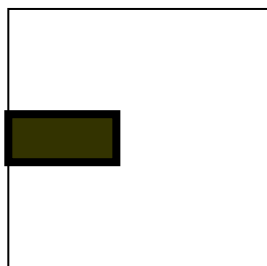


On a picture you can perform three operations: **grow()**, **split** and **die**.

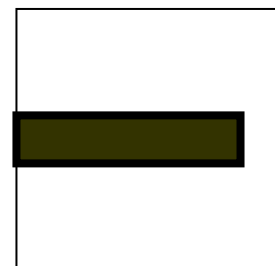
The following examples explain the semantics (and the syntax): each picture is obtained from the preceding as result of the operation in caption (the first picture is obtained from the empty picture).



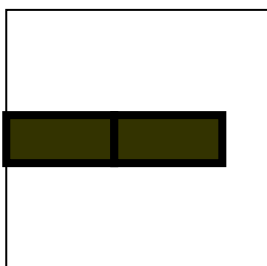
**a;**



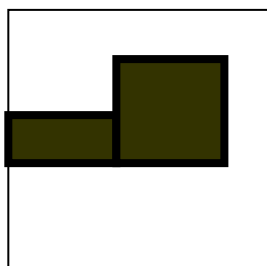
**a:grow(east);**



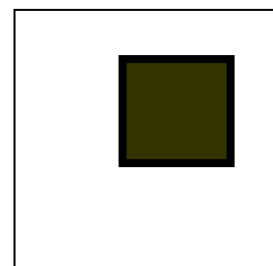
**a:grow(east);**



**b,c=a:split;**



**c:grow(north);**

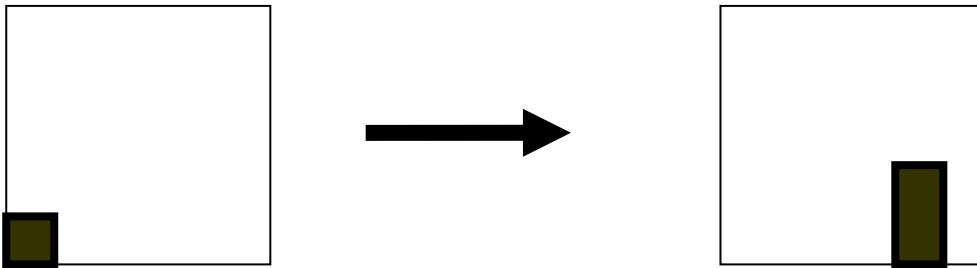


**b:die;**

Note that only on *oblong* object you may execute the split-operation producing two shorter objects of equal size. A *square* cannot be split.

PROBLEMA

Beaver wants to write a program that transforms the left image to the right image.



Which could be the first four commands of this program?

- A **a:grow(east); a:grow(east); b,c=a:split; b:die;**
- B **a:grow(north); a:grow(east); a:grow(east); b,c=a:split;**
- C **a:grow(east); a:grow(east); a:grow (north); a:die;**
- D **a:grow(east); b,c=a:split; c:grow(north); c:grow(east);**

Enter your answer as A, B, C or D in the box below.

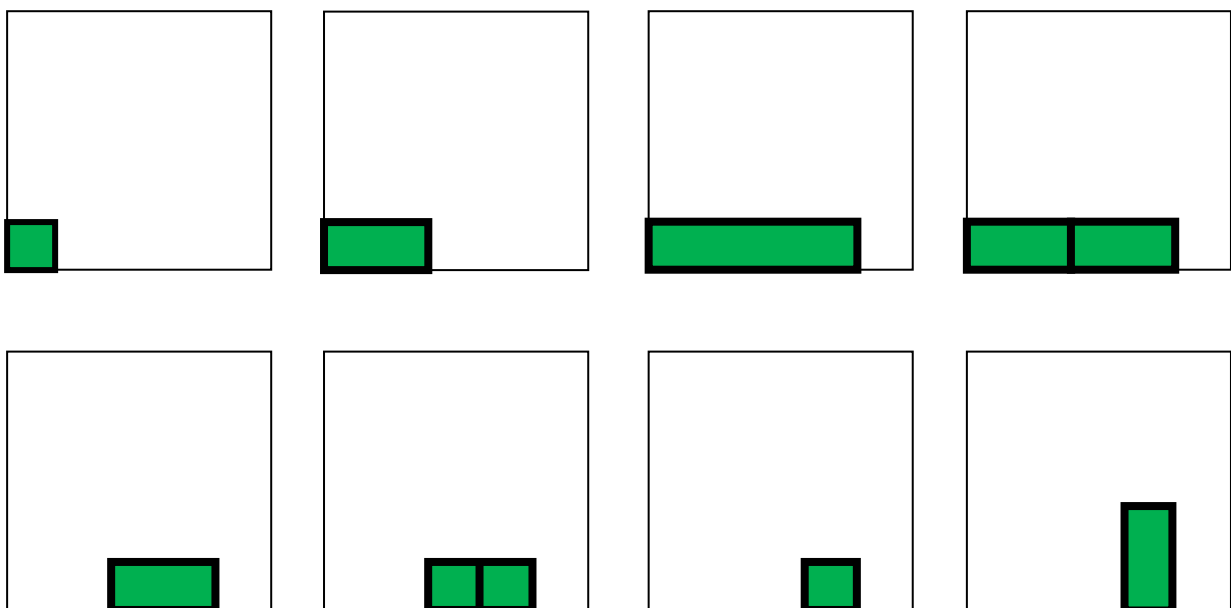
SOLUZIONE

COMMENTI ALLA SOLUZIONE

The correct solution is that only A can be completed. This is the complete program:

**a:grow(east); a:grow(east); b,c = a:split; b:die; d,e = c:split; d:die; e:grow(north);**

The step by step result of this program is as follow.



C does not work because after the fourth command the image is empty.

**Scuola secondaria di secondo grado - GARA 5 APR '12**

B and D result to rectangles (or squares) with minimum side lengths of two units. It is impossible to reduce such a shape to a rectangle with a minimum side length of one unit using the split-operation. A gardener must grow things carefully!