ESERCIZIO 1

PREMESSA
La relazione che lega il costo totale conoscendo quello unitario e il numero di oggetti acquistati può essere rappresentata col termine regola(<sigla>,[costo unitario, quantità], <costo totale>). Più in generale, con il termine

regola(<Sigla>,<Lista antecedenti>,<Conseguente>,<Peso>)

si può descrivere ogni regola di deduzione che consente di dedurre <Conseguente> conoscendo tutti gli elementi contenuti nella <Lista antecedenti>; ogni regola è identificata in modo univoco da <Sigla> e da un <Peso> che misura la difficoltà di applicazione di quella regola (per esempio, basso per una somma, più alto per una divisione). 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 ad esse corrispondenti. Ad ogni procedimento può essere associato un peso complessivo dato dalla somma dei pesi delle singole regole che lo compongono.

PROBLEMA
È dato il seguente insieme di regole (in cui il nome del termine è “r” invece di “regola”):

| | | | |
|---|---|---|---|
| r(1,[c1,c2],i,12). | r(2,[c1,i],c2,7). | r(3,[c2,i],c1,7). | r(4,[i,h],a,7). |
| r(5,[a,h],i,7). | r(6,[i,a],h,7). | r(7,[c1,c2],a,12). | r(8,[c1,a],c2,12). |
| r(9,[c2,a],c1,12). | r(10,[c1,p1],h,7). | r(11,[c1,h],p1,7). | r(12,[p1,h],c1,7). |
| r(13,[p1,p2],h,8). | r(14,[h,p1],p2,7). | r(15,[p2,h],p1,7). | r(16,[c2,p2],h,7). |
| r(17,[c2,h],p2,7). | r(18,[p2,h],c2,7). | | |

Si osserva che, conoscendo **c1** e **c2**, è possibile dedurre **i** con la regola 1 e **a** con la regola 7; ma è anche possibile dedurre **h** con la regola 6 dopo aver applicato prima la regola 1 (per dedurre **i**), poi la regola 7 (per dedurre **a**). Quindi, la lista [1,7,6] descrive un procedimento per dedurre **h** conoscendo [**c1**,**c2**]; il peso complessivo è 31.
Utilizzando le regole sopra riportate trovare i numeri N1 e N2 di regole che si devono complessivamente applicare, conoscendo **h** e **p2**, per dedurre **a** rispettivamente con la regola 4 e con la regola 7; determinare inoltre i pesi P1 e P2 dei rispettivi procedimenti.
N.B. Quando due regole possono essere applicate in alternativa, applicare per prima quella con la sigla di valore più basso.

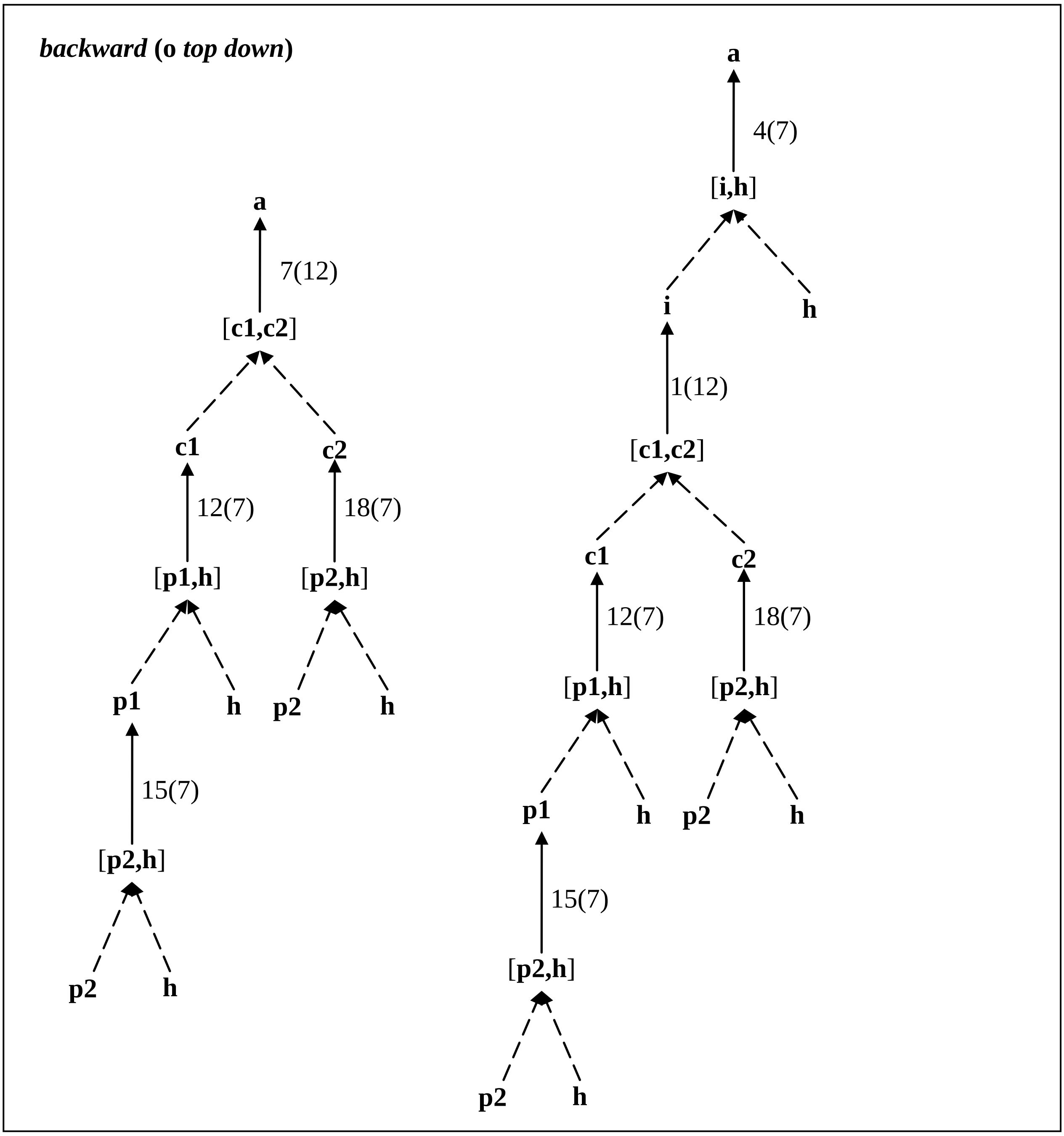
| | |
|---|---|
| N1 | |
| N2 | |
| P1 | |
| P2 | |

SOLUZIONE

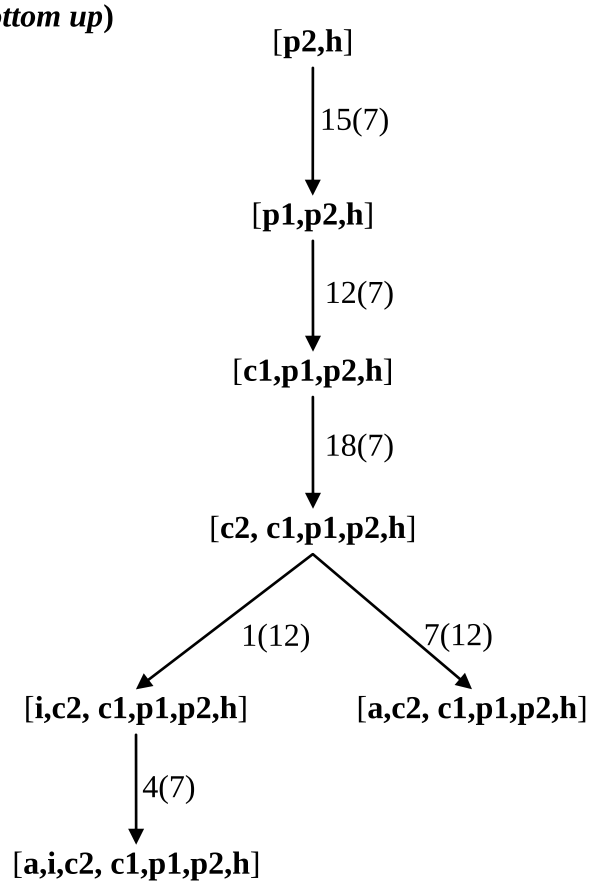
| | |
|---|---|
| N1 | 5 |
| N2 | 4 |
| P1 | 40 |
| P2 | 33 |

COMMENTI ALLA SOLUZIONE
Per dedurre **a** si possono usare due regole la 4 e la 7, quindi è possibile che ci siano almeno due procedimenti. La regola 4 può essere applicata a partire da [i,h]; h è noto, ma i non lo è, quindi si deve cercare una regola per dedurre i. Si trova la regola 1 (che richiede di conoscere [c1,c2]) e quindi si procede per dedurre c1 e c2 e si va avanti finché non si trova una regola applicabile con i dati del problema.

In generale, per risolvere il problema si può usare 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) la soluzione è trovata, altrimenti si continua a cercare regole per derivare i termini incogniti; il metodo è illustrato nella *prima* figura seguente, in cui le frecce non tratteggiate (di tipo OR) indicano le regole applicabili (la sigla è scritta a fianco) e le frecce tratteggiate (di tipo AND) indicano gli antecedenti della regola. In questo modo si trovano procedimenti per derivare l'incognita rappresentati graficamente da alberi, le cui foglie sono (tutte) dati.
I due alberi, mostrati in figura, descrivono i processi di derivazione richiesti da questo problema.
Un altro metodo è quello *forward* (o *bottom up*) che consiste nel partire dai dati e usare le regole applicabili per aumentare la conoscenza via via fino a comprendere l'incognita; il metodo è illustrato nella *seconda* figura seguente; anche in questo caso, naturalmente, si ottiene un albero che descrive un solo processo di derivazione.
N.B. Nel primo caso la successione delle regole applicate è dal basso verso l'alto; nel secondo caso è dall'alto al basso.
I procedimenti sono [15,12,18,1,4] e [15,12,18,7].



***forward* (o *bottom up*)**



ESERCIZIO 2

PREMESSA

In un foglio a quadretti è disegnato un campo di gara, per esempio di 14 quadretti in orizzontale e 5 in verticale (vedi figura).

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | |
| | | | | | | | | | S | | | | |
| | | | | | P | | | | | | | | |
| | | | | | | | | | | | | | |
| ➔ | | | | | | | | | | | | | |

Ogni casella può essere individuata da due numeri (interi); per esempio la casella contenente P è individuata da essere nella sesta colonna (da sinistra) e nella terza riga (dal basso): brevemente si dice che ha *coordinate* [6,3]; la prima coordinata (in questo caso 6) si dice *ascissa* e la seconda (in questo caso 3) si dice *ordinata*. Le coordinate della casella contenente S sono [10,4] e di quella contenente la freccia sono [1,1].

La freccia può essere pensata come un robot, in questo caso rivolto verso destra; il robot può eseguire tre tipi di comandi:

– girarsi di 90 gradi in senso *orario*: comando o;
– girarsi di 90 gradi in senso *antiorario*: comando a;
– avanzare di una casella (nel senso della freccia, mantenendo l'orientamento): comando f.

Questi comandi possono essere concatenati in sequenze in modo da permettere al robot di compiere vari percorsi; per esempio la sequenza di comandi descritta dalla lista [f,f,f,f,f,a,f,f] fa spostare il robot dalla posizione e orientamento iniziali mostrati in figura fino alla casella P; risultato analogo si ottiene con la lista [a,f,f,o,f,f,f,f,f]. Tuttavia, nel primo caso l'orientamento finale del robot è verso l'alto, mentre nel secondo caso l'orientamento finale è verso destra. Il robot ha sempre uno dei quattro orientamenti seguenti descritti con: n (nord, verso l'alto), s (sud, verso il basso), e (est, verso destra), o (ovest, verso sinistra).

N.B. Non confondere "o" come descrizione dell'orientamento e "o" come comando.

PROBLEMA

In un campo di gara sufficientemente ampio si trovano due robot che devono compiere due tragitti così definiti:

- primo robot: coordinate della partenza [9,9], direzione n, lista dei comandi:

[a,f,f,o,f,f,a,a,f,f,f,a,o,f,f,f,o,f,f,f,f];

- secondo robot: coordinate della partenza [10,7], direzione s, lista dei comandi:

[o,f,f,f,f,f,o,f,f,a,f,o,f,f,a,f,o,o,f,f,o].

Trovare le coordinate X1, Y1 e l'orientamento D1 del primo robot al termine del suo percorso.
Trovare le coordinate X2, Y2 e l'orientamento D2 del secondo robot al termine del suo percorso.
Trovare le coordinate X3, Y3 della casella comune ai due percorsi.

| | |
|---|---|
| X1 | |
| Y1 | |
| D1 | |
| X2 | |
| Y2 | |
| D2 | |
| X3 | |

| | |
|---|---|
| Y3 | |

SOLUZIONE

| | |
|---|---|
| X1 | 3 |
| Y1 | 5 |
| D1 | o |
| X2 | 5 |
| Y2 | 11 |
| D2 | s |
| X3 | 7 |
| Y3 | 7 |

COMMENTI ALLA SOLUZIONE

I percorsi dei robot si possono ottenere disegnandoli o, in maniera più sistematica, compilando tabelle come le seguenti, che mostrano lo stato del robot *dopo* ogni comando. (Si noti come l'azione del comando "f", che aumenta o diminuisce una della coordinate, dipende dall'orientamento.)

**PRIMO ROBOT**

| SITUAZIONE | STATO |
|---|---|
| partenza | [9,9,n] |
| 1 comando: a | [9,9,o] |
| 2 comando: f | [8,9,o] |
| 3 comando: f | [7,9,o] |
| 4 comando: o | [7,9,n] |
| 5 comando: f | [7,10,n] |
| 6 comando: f | [7,11,n] |
| 7 comando: a | [7,11,o] |
| 8 comando: a | [7,11,s] |
| 9 comando: f | [7,10,s] |
| 10 comando:f | [7,9,s] |
| 11 comando: f | [7,8,s] |
| 12 comando: a | [7,8,e] |
| 13 comando: o | [7,8,s] |
| 14 comando: f | [7,7,s] |
| 15 comando: f | [7,6,s] |
| 16 comando: f | [7,5,s] |
| 17 comando: o | [7,5,o] |
| 18 comando: f | [6,5,o] |
| 19 comando: f | [5,5,o] |
| 20 comando: f | [4,5,o] |
| 21 comando: f | [3,5,o] |
| arrivo | [3,5,o] |

**SECONDO ROBOT**

| SITUAZIONE | STATO |
|---|---|
| partenza | [10,7,s] |
| 1 comando: o | [10,7,o] |
| 2 comando: f | [9,7,o] |

| | |
|---|---|
| 3 comando: f | [8,7,o] |
| 4 comando: f | [7,7,o] |
| 5 comando: f | [6,7,o] |
| 6 comando: f | [5,7,o] |
| 7 comando: o | [5,7,n] |
| 8 comando: f | [5,8,n] |
| 9 comando: f | [5,9,n] |
| 10 comando:a | [5,9,o] |
| 11 comando: f | [4,9,o] |
| 12 comando: o | [4,9,n] |
| 13 comando: f | [4,10,n] |
| 14 comando: f | [4,11,n] |
| 15 comando: a | [4,11,o] |
| 16 comando: f | [3,11,o] |
| 17 comando: o | [3,11,n] |
| 18 comando: o | [3,11,e] |
| 19 comando: f | [4,11,e] |
| 20 comando: f | [5,11,e] |
| 21 comando: o | [5,11,s] |
| arrivo | [5,11,s] |

L'unica casella comune ai due percorsi è [7,7].

ESERCIZIO 3

PROBLEMA

Nel seguente testo sostituire a X1, X2, X3, ecc. la parola più appropriata, scelta tra quelle proposte. (N.B. solo una scelta è *coerente* col significato generale del testo, anche se altre sono sintatticamente possibili; per svolgere l'esercizio non è necessario conoscere l'argomento trattato nel brano).

I X1 domestici arrivarono in Europa da oriente, durante l'ottavo secolo a. C. Introdotti inizialmente in Grecia, continuarono a costituire una X5 per più di un secolo; Omero ed Esiodo, che vissero entrambi dopo l'ottocento e prima del settecento a. C. non li conoscevano. Il primo riferimento di un X6 a questi uccelli si trova negli scritti del poeta gnomico elegiaco Teognide, nato a Megara verso la metà del sesto secolo a. C. Sul finire dello stesso secolo il X2 diventa un elemento comunissimo della vita economica umana e la sua natura bellicosa serve da paradigma ai moralisti nelle loro lezioni. Il X3 della commedia greca, Epicarmo, forse originario di Coo ma vissuto per moltissimi anni a Siracusa, sembra aver fatto molti riferimenti al X2. Il X2 è nominato anche da Pindaro (518-438 a. C.), che paragona le disonorevoli vittorie della guerra civile alle lotte che si svolgono nel X4.

Lista delle scelte:

| | |
|---|---|
| A ruminanti | M figura |
| B rarità | N greco |
| C pozzo | O latino |
| D scarsità | P gallo |
| E gallinacei | Q suini |
| F apparenza | R bue |
| G figlio | S speranza |
| H ovini | T padre |
| I pollaio | U bovini |
| L fenicio | V vicolo |

Indicare le scelte con la lettera maiuscola corrispondente.

| | |
|---|---|
| X1 | |
| X2 | |
| X3 | |
| X4 | |
| X5 | |
| X6 | |

SOLUZIONE

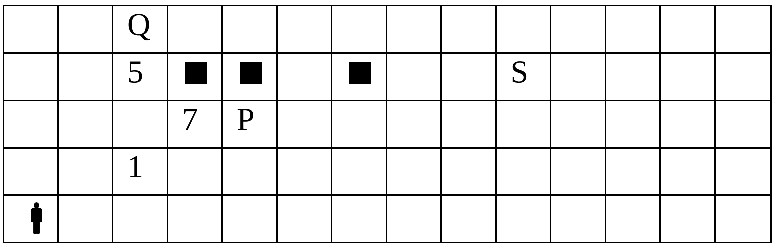
| | |
|---|---|
| X1 | E |
| X2 | P |
| X3 | T |
| X4 | I |
| X5 | B |
| X6 | N |

COMMENTI ALLA SOLUZIONE

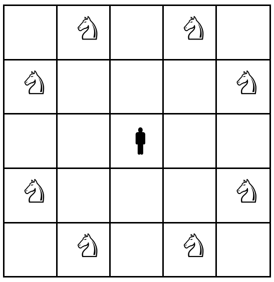
| Variabile | Presumibili proprietà grammaticali o sintattiche | Scelte possibili | Scelta corretta |
|---|---|---|---|
| X1 | sostantivo plurale maschile | ruminanti, gallinacei, ovini, suini, bovini | gallinacei (più naturale nel contesto) |
| X2 | sostantivo maschile | pozzo, figlio, pollaio, fenicio, greco, latino, gallo, bue, padre, vicolo | gallo (unica scelta coerente col contesto) |
| X3 | sostantivo maschile | pozzo, figlio, pollaio, fenicio, greco, latino, gallo, bue, padre, vicolo | padre (unica scelta naturale nel contesto) |
| X4 | sostantivo maschile | pozzo, figlio, pollaio, fenicio, greco, latino, gallo, bue, padre, vicolo | pollaio (unica scelta naturale nel contesto) |
| X5 | sostantivo femminile | rarità, scarsità, apparenza, figura, speranza | rarità (più naturale nel contesto) |
| X6 | sostantivo maschile | pozzo, figlio, pollaio, fenicio, greco, latino, gallo, bue, padre, vicolo | greco (deve essere coerente col contesto geografico) |

ESERCIZIO 4

PREMESSA
In un foglio a quadretti è disegnato un campo di gara di dimensioni 14×5 (14 quadretti in orizzontale e 5 in verticale, vedi figura).




Ogni casella può essere individuata da due numeri (interi); per esempio la casella contenente P è individuata da essere nella quinta colonna (da sinistra) e nella terza riga (dal basso): brevemente si dice che ha *coordinate* [5,3]; la prima coordinata (in questo caso 5) si dice *ascissa* e la seconda (in questo caso 3) si dice *ordinata*. Le coordinate della casella contenente S sono [10,4] e di quella contenente il robot  sono [1,1].
Il robot si muove a passi e ad ogni passo (o mossa) può spostarsi solo in una delle caselle contenenti ♘ come illustrato nella seguente figura (allo stesso modo del *cavallo* nel gioco degli scacchi).



Il campo di gara contiene caselle interdette al robot (segnate da un quadrato nero in figura) quindi, tenuto conto anche dei bordi del campo di gara, la mobilità del robot può essere limitata; ad esempio se il robot si trovasse nella casella in cui c'è Q si potrebbe spostare solo in 3 caselle; se fosse nella casella in cui c'è P avrebbe 7 mosse possibili; dalla casella [1,1] ha solo 2 mosse possibili.
In alcune caselle sono posti dei premi che il robot può accumulare lungo un percorso. I premi sono descritti fornendo le coordinate della casella che lo contiene e il valore del premio: i premi sopra riportati sono descritti dalla seguente lista [[3,2,1],[4,3,7],[3,4,5]]. Un percorso è descritto dalla lista delle coordinate delle caselle attraversate. Un possibile percorso da P (coordinate [5,3]) a Q (coordinate [3,5]) è descritto dalla seguente lista: [[5,3],[3,2],[5,1],[4,3],[3,5]] e ha un totale di premi accumulati pari a 8.

PROBLEMA
In un campo di gara di dimensioni 4×4 il robot si trova nella casella [4,4]; deve fare un percorso *chiuso*, cioè partire dalla casella iniziale e ritornarci, e *semplice*, cioè senza passare due volte in una stessa casella (quindi tutte le caselle del percorso sono diverse, tranne la partenza e l'arrivo); nel campo sono presenti le caselle interdette descritte dalla seguente lista:

[[1,4],[2,1],[3,4],[4,1],[2,4]].

I premi distribuiti nel campo di gara sono descritti dalla seguente lista:

[[1,3,3],[1,2,5],[3,3,9],[3,2,6],[3,1,1]].

Trovare:

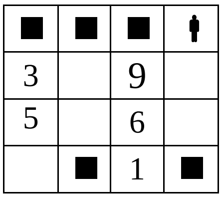
- il numero N di possibili percorsi diversi chiusi e semplici;
- la lista L che descrive il percorso che consente di non raccogliere premi.

| | |
|---|---|
| N | |
| L | |

Soluzione

| | |
|---|---|
| N | 4 |
| L | [[4,4],[2,3],[4,4]] |

COMMENTI ALLA SOLUZIONE
Il campo di gara è mostrato nello schema seguente:




I percorsi possibili sono:

1. [[4,4],[2,3],[1,1],[3,2],[4,4]] premi accumulati 6
2. [[4,4],[2,3],[4,4]] premi accumulati 0
3. [[4,4],[3,2],[1,1],[2,3],[4,4]] premi accumulati 6
4. [[4,4],[3,2],[4,4]] premi accumulati 6

ESERCIZIO 5

PREMESSA
Alcuni ragazzi decidono di costruire un ipertesto multimediale sugli avvenimenti storici significativi della loro regione. Per organizzare il progetto, dividono il lavoro in singole attività e assegnano ogni attività a un gruppo di loro.
Le attività sono descritte col seguente termine

a(<sigla attività>,<durata in giorni>,<ragazzi impegnati>);

esempio, il termine a(A1,1,6) significa che l'attività A1 dura un giorno e impiega 6 ragazzi.
Le attività non possono svolgersi tutte contemporaneamente, ma devono essere rispettate delle priorità descritte con termini del tipo

p(<precedente>,<successiva>);

come per esempio p(A4,A8) e p(A6,A8); ogni termine esprime il fatto che l'attività associata alla sigla di destra (detta successiva) può iniziare solo quando l'attività associata alla sigla di sinistra (detta precedente) è terminata. Ovviamente se una attività ha più precedenti, può iniziare solo quando *tutte* le precedenti sono terminate; i due termini appena visti implicano che l'attività A8 può iniziare solo dopo che sono terminate le due attività A4 e A6.

PROBLEMA
Le attività di un progetto sono descritte nella seguente lista di termini:

[a(A1,1,2),a(A2,2,4),a(A3,3,3),a(A4,2,1),a(A5,2,1),a(A6,2,3),a(A7,2,3),a(A8,2,2),a(A9,2,5),a(A10,1,4),a(A11,1,3),a(A12,1,2),a(A13,2,3),a(A14,1,2),a(A15,2,4)].

Le priorità sono descritte dalla seguente lista di termini:

[p(A1,A2),p(A1,A3),p(A2,A4),p(A2,A5),p(A3,A6),p(A3,A7),p(A4,A8),p(A5,A8),p(A6,A11),p(A7,A10),p(A8,A9),p(A10,A15),p(A11,A10),p(A3,A5),p(A9,A15),p(A1,A12),p(A12,A14),p(A12,A7),p(A14,A13),p(A13,A10)].

Si supponga che ogni attività inizi *prima possibile* (nel rispetto delle priorità): determinare il numero N di giorni necessari per completare il progetto. Inoltre:

1) trovare il numero T1 del giorno in cui lavora il maggior numero RM di ragazzi;
2) trovare il numero T2 del giorno in cui lavora il minor numero Rm di ragazzi;
3) trovare il numero massimo AP di attività che si svolgono in parallelo;
4) il numero medio MG dei ragazzi che giornalmente lavorano al progetto (numero troncato a due cifre decimali).

N.B. I giorni sono numerati a partire dal primo giorno del progetto.

| | |
|---|---|
| N | |
| T1 | |
| RM | |
| T2 | |
| Rm | |
| AP | |
| MG | |

SOLUZIONE

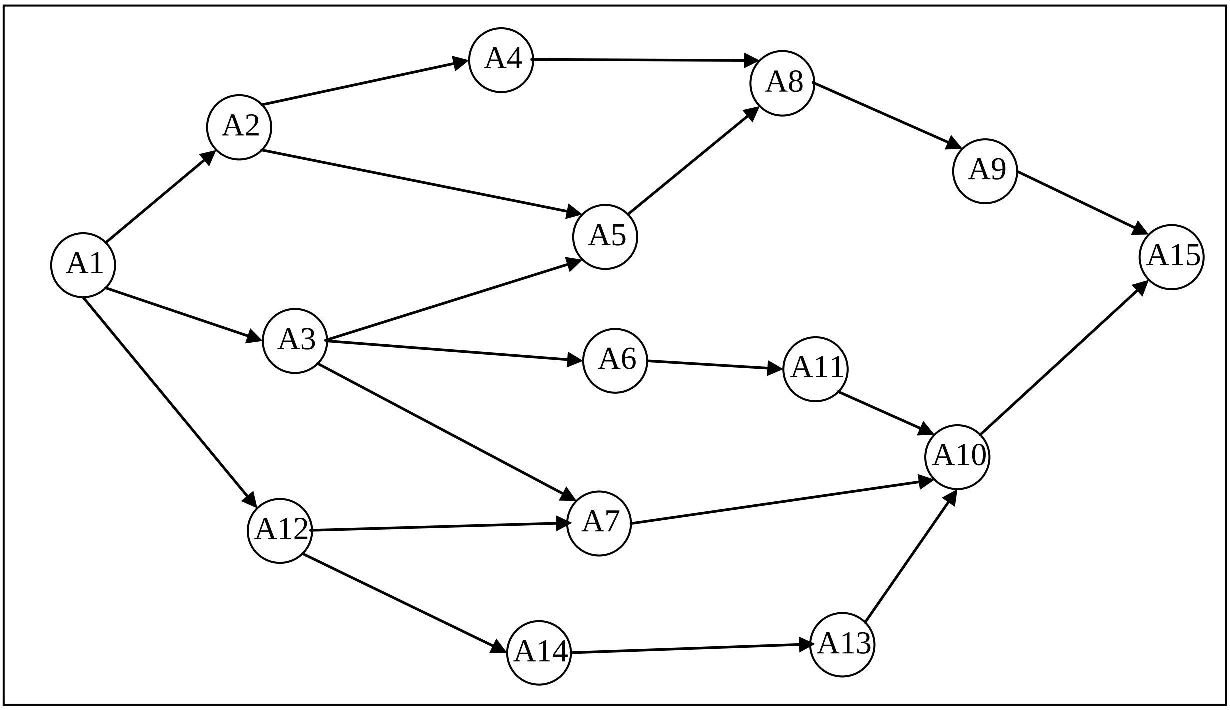
| | |
|---|---|
| N | 12 |
| T1 | 5 |
| RM | 11 |
| T2 | 1 |
| Rm | 2 |
| AP | 5 |
| MG | 6,16 |

COMMENTI ALLA SOLUZIONE

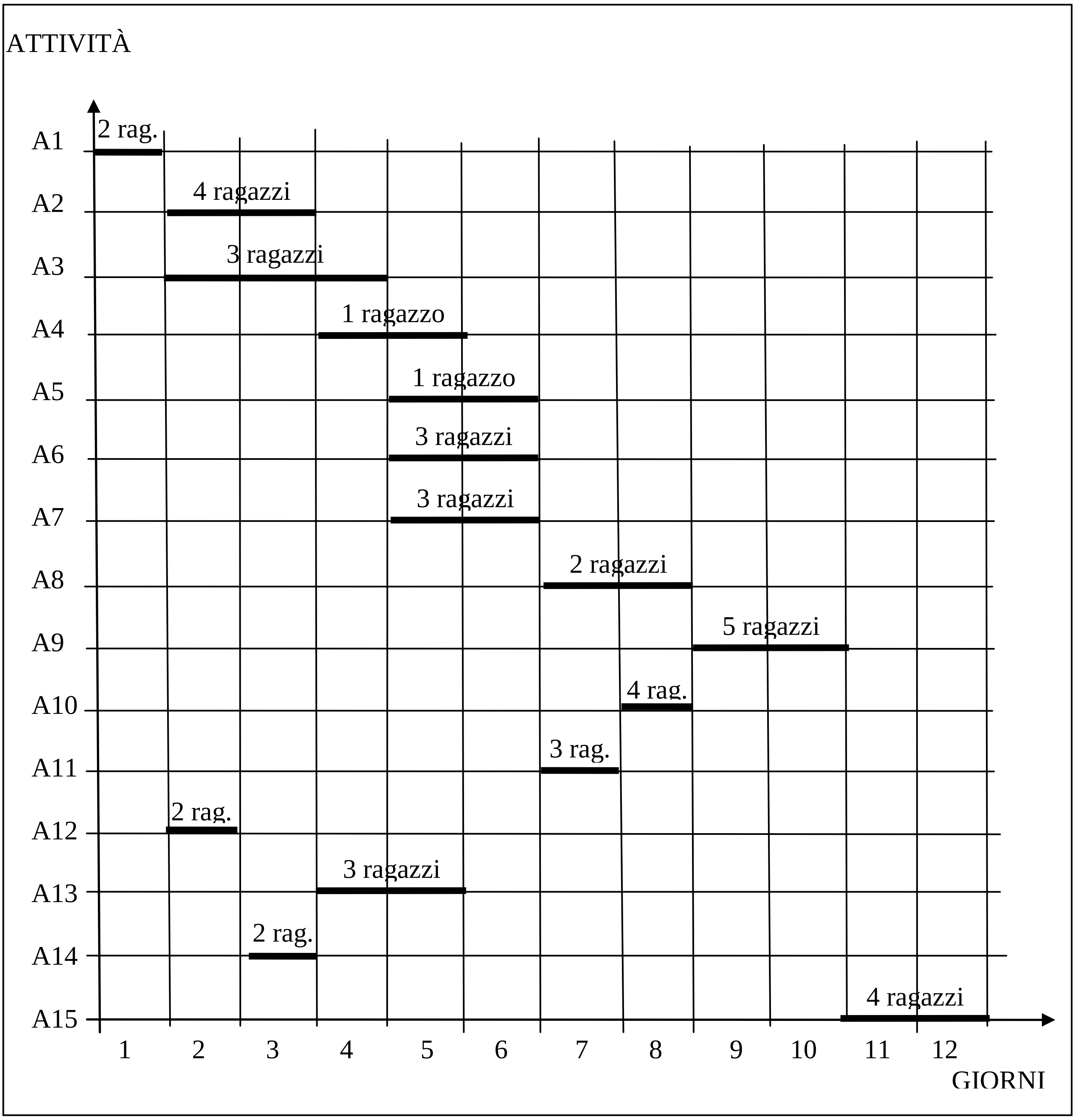
Per facilitare la soluzione è utile trasformare in tabella la lista che descrive la durata e le persone relative ad ogni attività.

| Attività | Durata | Ragazzi |
|---|---|---|
| A1 | 1 | 2 |
| A2 | 2 | 4 |
| A3 | 3 | 3 |
| A4 | 2 | 1 |
| A5 | 2 | 1 |
| A6 | 2 | 3 |
| A7 | 2 | 3 |
| A8 | 2 | 2 |
| A9 | 2 | 5 |
| A10 | 1 | 4 |
| A11 | 1 | 3 |
| A12 | 1 | 2 |
| A13 | 2 | 3 |
| A14 | 1 | 2 |
| A15 | 2 | 4 |

Successivamente è bene disegnare il diagramma delle precedenze, cioè il grafo che ha come nodi le attività e come frecce le precedenze. Si procede per passi successivi: prima si disegnano in "ordine sparso" dei nodi con etichette che sono le attività (per esempio un cerchietto con la sigla della attività); poi si congiungono con una freccia i nodi che appartengono a un elemento (sottolista) della lista che rappresenta le priorità; successivamente si procede a ridisegnare, per tentativi, il grafo cercando di "disintrecciare" le frecce: di solito ci riesce completamente, come nell'esempio in figura (ma non sempre è possibile). Si noti che esiste una "prima attività" del progetto (A1 in figura) e una "ultima attività" (A15 in figura).
N.B. È casuale che la prima attività abbia la sigla più piccola e l'ultima la sigla più grande.
Il diagramma delle precedenze esprime in maniera molto "leggibile" la precedenza tra le attività e consente di passare con facilità allo stadio successivo.




Dal grafo e dalla tabella si può compilare il Gantt; si elencano le attività sull'asse verticale, avendo cura di iniziare (dall'alto) con la prima attività e finire in basso con l'ultima; sull'asse orizzontale si elencano i giorni: *a priori* non si può dire quanti saranno necessari (certamente non più della somma di quelli che compaiono nella tabella che descrive le attività).




N.B. In casi complicati è bene riordinare le attività sull'asse verticale in modo che i segmenti che le rappresentano si spostano "con regolarità" da sinistra in alto verso destra in basso. Nella figura non è stato fatto.

Dal Gantt si deduce facilmente la lista delle coppie [giorno,persone]:

[[1,2],[2,9],[3,9],[4,7] ,[5,11],[6,7],[7,5],[8,6] ,[9,5],[10,5],[11,4],[12,4]].

Da tale lista viene immediatamente la soluzione: N vale 12; T1 vale 5 e RM vale 11; T2 vale 1 e Rm vale 2; il numero massimo di attività che si svolgono in parallelo è 5 (il giorno 5). Per calcolare la media dei ragazzi che giornalmente lavorano al progetto si sommano i ragazzi che lavorano ogni giorno e la somma si divide per N:

$$(2+9+9+7+11+7+5+6+5+5+4+4)/12 = 74/12 = 6{,}16.$$

ESERCIZIO 6

PREMESSA
Nello pseudolinguaggio oltre al tipo “integer” che descrive variabili che hanno valore intero, esiste anche il tipo “real”, che descrive variabili che hanno valore razionale: un tale valore si può pensare come un numero (in rappresentazione decimale) “con il punto”. In alcuni linguaggi di programmazione è usato il termine “float” invece di “real”.

N.B. Si segue la convenzione anglosassone di scrivere i numeri decimali col “.” *e non con la* “,”.

Il seguente programma illustra alcuni esempi di utilizzo di variabili di tipo “integer” e “real”.

```
Procedure ESEMPIO
variables I, N, M integer;
variables A, B, C, D real;
I = 2;
M = 5;
A = 2.0;
B = 5.0;
N = M/I;
C = B/A;
D = 3×(B+I)/M;
output N, C, D;
```

Al termine della esecuzione si ha: N = 2, C = 2.5, D = 4.2; se in una espressione sono presenti variabili “integer” e variabili “real”, *tutte* le valutazioni delle operazioni sono di tipo “real”.

PROBLEMA
Si consideri la seguente procedura.

```
procedure PROVA1;
variables N, S, I integer;
variables T real;
input N;
S ← 20;
T ← 1.0;
for I from 1 to N step 1 do
        S ← S + 3 × I + 2;
        T ← T + S / 5.0 + I + 1.0;
endfor;
output S, T;
endprocedure;
```

N.B. Osservare che T è di tipo “real”, cioè è un numero razionale; il suo valore ha sempre (almeno) una cifra decimale, separata dalla parte intera da un punto.

Compresa la sequenza dei calcoli descritti, eseguire le operazioni indicate sapendo che il valore in input per N è 8; trovare i valori di output per S e T e il più piccolo valore di I per cui risulta T>S.

| | |
|---|---|
| S | |
| T | |
| I | |

SOLUZIONE

| | |
|---|---|
| S | 144 |
| T | 163.4 |
| I | 7 |

COMMENTI ALLA SOLUZIONE
Il ciclo “for” viene ripetuto 8 volte; i valori delle variabili prima del ciclo e dopo ogni ripetizione sono riportati nella seguente tabella.

| | **N** | **I** | **S** | **T** |
|---|---|---|---|---|
| valori immediatamente prima del ciclo “for” | 8 | / | 20 | 1.0 |
| valori dopo la ripetizione 1 del ciclo “for” | 8 | 1 | 25 | 8.0 |
| valori dopo la ripetizione 2 del ciclo “for” | 8 | 2 | 33 | 17.6 |
| valori dopo la ripetizione 3 del ciclo “for” | 8 | 3 | 44 | 30.4 |
| valori dopo la ripetizione 4 del ciclo “for” | 8 | 4 | 58 | 47.0 |
| valori dopo la ripetizione 5 del ciclo “for” | 8 | 5 | 75 | 68.0 |
| valori dopo la ripetizione 6 del ciclo “for” | 8 | 6 | 95 | 94.0 |
| valori dopo la ripetizione 7 del ciclo “for” | 8 | 7 | 118 | 125.6 |
| valori dopo la ripetizione 8 del ciclo “for” | 8 | 8 | 144 | 163.4 |

N.B. La variabile I non ha valore prima del ciclo for” (come del resto le altre variabili all’inizio della procedura).

ESERCIZIO 7

PREMESSA
La struttura
```
for I from 1 to N step 1 do
    <ciclo>
endfor;
```
prescrive di ripetere le azioni contenute in <ciclo> N volte con I = 1, 2, 3, ... N.
La ripetizione delle azioni può essere descritta anche con la seguente struttura:
```
while <condizione> do
    <ciclo>
endwhile;
```
In questa struttura <condizione> deve essere sostituita da una espressione che può essere vera o falsa, per esempio A>B. In tale esempio si richiede che prima di iniziare il ciclo, devono essere assegnati valori ad A e B; se è vero che A>B allora il <ciclo> viene eseguito; terminato il ciclo, si torna a valutare la <condizione>. Nel ciclo i valori di A o di B (o di entrambi) devono cambiare, altrimenti il ciclo verrebbe ripetuto all'infinito. Quando al termine di un ciclo la <condizione> è falsa (nell'esempio non risulta più A>B), il ciclo non viene ripetuto e il calcolo passa alla istruzione successiva a endwhile.

PROBLEMA
Compresa la sequenza dei calcoli descritti nella procedura seguente, eseguire le operazioni indicate utilizzando i dati di input sotto riportati e trovare i valori di output.

```
procedure PROVA2;
variables N, S, T integer;
input S,T;
N ← 0;
while S>T do
    N ← N+ 1;
    S ← S + N^2 + 1;
    T ← T – S + N^3;
endwhile;
output N, S, T;
endprocedure;
```

In input si ha il valore 30 per S e il valore 1 per T.

| | |
|---|---|
| S | |
| T | |
| N | |

SOLUZIONE

| | |
|---|---|
| S | 177 |
| T | 211 |
| N | 7 |

4 ?- p(30,1).
t(1, 32, -30)
t(2, 37, -59)

t(3, 47, -79)
t(4, 64, -79)
t(5, 90, -44)
t(6, 127, 45)
t(7, 177, 211)

COMMENTI ALLA SOLUZIONE
Prima del ciclo “while” è vero che S>T (dati i valori di input per S e T), quindi il ciclo viene eseguito; alla fine viene rivalutata la condizione S>T e il ciclo viene eseguito se la condizione è vera, altrimenti si passa a eseguire le azioni successive.
*A priori* non è possibile stabilire quante volte è ripetuto il ciclo; nella procedura PROVA2 il numero delle ripetizioni è “contato” dal valore della variabile N.
I valori delle variabili prima del ciclo e dopo ogni ripetizione sono riportati nella seguente tabella.

| | **N** | **S** | **T** |
|---|---|---|---|
| valori prima del ciclo “while” | 0 | 30 | 1 |
| valori dopo la ripetizione 1 del ciclo “while” | 1 | 32 | -30 |
| valori dopo la ripetizione 2 del ciclo “while” | 2 | 37 | -59 |
| valori dopo la ripetizione 3 del ciclo “while” | 3 | 47 | -79 |
| valori dopo la ripetizione 4 del ciclo “while” | 4 | 64 | -79 |
| valori dopo la ripetizione 5 del ciclo “while” | 5 | 90 | -44 |
| valori dopo la ripetizione 6 del ciclo “while” | 6 | 127 | 45 |
| valori dopo la ripetizione 7 del ciclo “while” | 7 | 177 | 211 |

ESERCIZIO 8

PROBLEMA

Una azienda ha costruito un nuovo motore sperimentale molto potente che, però, ha consumi elevati. Vengono apportate tre modifiche: la prima permette un risparmio di carburante del 30%, la seconda del 45% e la terza del 25%. Calcolare il risparmio R complessivo (supponendo, naturalmente, che le tre modifiche possono funzionare contemporaneamente e in maniera indipendente).
N.B. Nella risposta usare 3 cifre decimali a destra della virgola.

| | |
|---|---|
| R | |

SOLUZIONE

| | |
|---|---|
| R | 71,125 |

COMMENTI ALLA SOLUZIONE

Conviene fissare l'attenzione sui consumi (per unità di tempo o di percorso, con diversa unità di misura). Se il motore non modificato consuma 100, con la prima modifica consuma 70 (il 30% di risparmio); con anche la seconda modifica consuma 70 × (100 – 45) / 100 = 38,50; con anche la terza modifica consuma 38,50 × (100 –25) / 100 = 28,8750
Quindi il risparmio sul consumo originario è del (100 – 28,8750)% = 71,125%

ESERCIZIO 9

PROBLEMA

Suppose that each *step* can be of type

a: add 1

b: multiply by 2.

Starting at 1, the steps to get to 9 could be described by a list of "a" and "b"

[b,b,b,a], [a,a,a,a,a,a,a,a,], [a,b,b,a], …

Note that the first step can always be either "a" or "b". Anyway the *minimum* number of step is 4.
What is the minimum number of steps to get to 51?
Enter your answer in the box below.

| |
|---|
| |

SOLUZIONE

| |
|---|
| 8 |

COMMENTI ALLA SOLUZIONE

For 51 you need 8 steps [b,a,b,b,b,a,b,a] (or [a,a,b,b,b,a,b,a]):

1*2 = 2
2+1 = 3
3*2 = 6
6*2 = 12
12*2 = 24
24+1 = 25
25*2 = 50
50+1 = 51

It's easy if you work *backward*; 51 is odd so the last step must be "a" from 50 which is even; the preceding step must be "b" from 25 which is odd; and so on.

ESERCIZIO 10

PREMESSA

Here are two simple arithmetical series of *five* numbers each; the differences are therefore equal between successive numbers in the same series - but not (necessarily!) the same in both series, as you may see.

7 13 19 25 31 and 7 12 17 22 27

But if the ten numbers are hideously jumbled, how to split them into their two series? Well, first list the numbers sequentially (i.e. in non decreasing order):

[7,7,12,13,17,19,22,25,27,31]

Now, the greatest number (31) must be the last in a series, so that the difference between 31 and one of the preceding numbers must be divisible by four (because of *four* gaps between *five* numbers). This pinpoints the first number of that series, always providing that all of the needed intermediate numbers are present.

If the ten number list is marked off with the found series, the other series is automatically revealed:

[7,7,12,13,17,19,22,25,27,31]

Summarising; the difference between 31 and 7 (i.e. 24), being divisible by four, offers us a series in increments of six which we know to be one of the series. As already stated, the other series is a straight read-off, from the list of ten, of the other five numbers.

PROBLEMA

Determine the two series in the following cases

A) [2,7,12,13,17,18,22,23,28,33]
B) [11,17,18,23,24,29,30,35,36,42]
C) [17,19,20,22,23,25,26,28,29,31]
D) [22,26,30,31,34,34,37,38,40,43]

Enter your answer in form of two lists in the table below.
N.B. Enter first the list to which belongs the last element of the given list.

| | | |
|---|---|---|
| A | | |
| B | | |
| C | | |
| D | | |

SOLUZIONE

| | | |
|---|---|---|
| A | [13,18,23,28,33] | [2,7,12,17,22] |
| B | [18,24,30, 36,42] | [11,17,23,29,35] |
| C | [19,22,25,28,31] | [17,20,23,26,29] |
| D | [31,34,37,40,43] | [22,26,30,34,38] |

COMMENTI ALLA SOLUZIONE

A) [2,7,12,13,17,18,22,23,28,33] marking [2,7,12,13,17,18,22,23,28,33]
B) [11,17,18,23,24,29,30,35,36,42] marking [11,17,18,23,24,29,30,35,36,42]
C) [17,19,20,22,23,25,26,28,29,31] marking [17,19,20,22,23,25,26,28,29,31]
D) [22,26,30,31,34,34,37,38,40,43] marking [22,26,30,31,34,34,37,38,40,43]