

INFORMATICA GENERALE

Homework 3: Scacchi, che passione!

docente: IVANO SALVO
Sapienza Università di Roma

pubblicazione: 10.V.2012 - consegna 23.V.2012

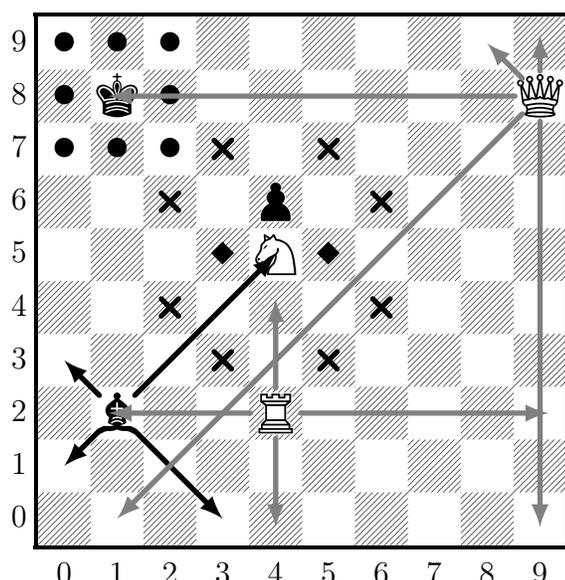


Figura 1: Movimento dei pezzi degli scacchi.

La Figura 1 ha l'ambizione di riassumere i movimenti di tutti i pezzi degli scacchi¹. Ogni pezzo può muovere dalla casella in cui si trova in un'altra casella della scacchiera. Se un pezzo P può muovere in una certa casella c , diremo che c è *attaccata* da P . Il Re (nero, casella [8,1]) si muove su caselle adiacenti in tutte le direzioni (pallini neri in figura). L'alfiere (nero, casella [2,1]) si muove lungo le diagonali (frecce nere). La torre (bianca, casella [2,4]) si muove lungo righe e colonne (frecce grigie). Il movimento di torre e alfiere si arresta quando un altro pezzo ne ostacola il movimento. Se il pezzo è un pezzo avversario, questo può essere *catturato* (o *preso*) ed eliminato dal gioco. Nel nostro esempio, la torre bianca può catturare l'alfiere nero (occupandone la posizione), e l'alfiere nero può catturare il cavallo bianco. Il cavallo (casella [5,4]) si muove con un caratteristico andamento ad "L" percorrendo una casella in una direzione (orizzontale o verticale), e due nell'altra

¹La numerazione ha il solo scopo di riferire i pezzi nel seguito. Nei vostri programmi rappresentate la scacchiera con una matrice $n \times n$ numerando righe e colonne da 0 a $n - 1$ e nell'esercizio 1 stampate la matrice partendo da riga 0.

Input:

```
6
0
0
```

Output:

```
0 3 2 3 2 3
3 4 1 2 3 4
2 1 4 3 2 3
3 2 3 2 3 4
2 3 2 3 4 3
3 4 3 4 3 4
```

Input:

```
3
1
0
```

Output:

```
3 2 1
0 -1 4
3 2 1
```

Figura 2: Esempi di input-output Esercizio 1.

(caselle contrassegnate con una 'x'). Cattura come gli altri pezzi, ma a differenza di torre e alfiere, il suo movimento non viene intralciato da eventuali altri pezzi sulla sua traiettoria (ad esempio il pedone nero in figura). Il pedone, che può andare solo avanti, si muove lungo la colonna, ma attacca le due caselle adiacenti in diagonale (rombi neri in figura) e può catturare pezzi avversari che si trovino lì. Infine, la donna (bianca, casella [8,9]) ha il movimento combinato di torre e alfiere. Diremo, inoltre che il re è *sotto scacco* se si trova in una casella attaccata da un pezzo nemico. Nel nostro caso, il re nero è sotto scacco a causa della donna bianca.

Esercizio 1: [CAMMINI MINIMI DEL CAVALLO] Il cavallo spesso impiega molte mosse anche per raggiungere caselle adiacenti. Ad esempio, può raggiungere le caselle nere a lui adiacenti in figura solo dopo almeno 3 mosse.

Voi dovete scrivere un programma che legge 3 interi, n , i e j con $0 \leq i, j < n$: e interpreta n come la dimensione della scacchiera, la coppia $[i, j]$ come la posizione del cavallo.

Il programma dovrà produrre in output una matrice $n \times n$ (stampata per righe) in cui c'è 0 in posizione $[i, j]$ e in ogni altra casella $[i', j']$ c'è il numero *minimo* di mosse necessario ad un cavallo per raggiungere $[i', j']$ partendo dalla casella $[i, j]$, e -1 se la casella non è raggiungibile (questo accade per scacchiere molto piccole). Stampate ogni elemento $a[i, j]$ della matrice con l'istruzione

`printf("%3d",a[i,j]);` e supponete che la matrice abbia massima dimensione 20. In Figura 2 sono mostrati due esempi di input-output.

ATTENZIONE: per ottenere risultati corretti, dovete prima calcolare *tutte* le caselle raggiungibili in un passo, poi partendo da questo *tutte* le caselle raggiungibili in 2 passi e così via, finchè non ci sono più nuove caselle raggiungibili.

Esercizio 2: [SCACCO DI REGINA] Scrivere un programma C che legge 5 interi, n, i, j, k ed l con $0 \leq i, j, k, l < n$: e interpreta n come la dimensione della scacchiera, la coppia $[i, j]$ come la posizione di una donna e la coppia $[k, l]$ come la posizione di un re (avversario).

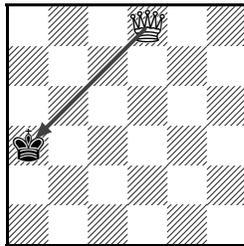
Il programma dovrà produrre in output 0 se il re è sotto scacco (si trova cioè in una casella attaccata dalla donna) e un intero n altrimenti, dove n è il numero delle possibili mosse di donna che danno scacco al re (essendo la scacchiera libera, è sempre possibile che la donna possa dare scacco al re).

Nelle prossime due figure abbiamo esemplificato possibili input e output. Nel diagramma corrispondente è visualizzata la posizione dell'input, e il motivo dell'output (la freccia indica come la donna dà scacco, mentre le 'x' indicano le caselle dove la donna può portarsi per dare scacco).

Input:

6
0
3
3
0

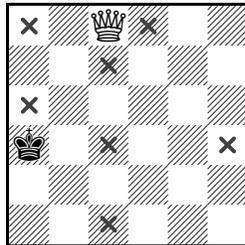
Output: 0



Input:

6
0
2
3
0

Output: 7



***Esercizio 3:** (VERIFICA DELLO SCACCO AL RE)². Una scacchiera (in questo esercizio consideriamo scacchiere tradizionali di lato 8) è rappresentata da 8 stringhe di 8 caratteri. Una stringa può contenere i caratteri: T, t, C, c, A, a, D, d, R, r, P, p, ., dove:

- . rappresenta una casella vuota;
- le lettere maiuscole rappresentano caselle occupate pezzi bianchi e le lettere minuscole rappresentano pezzi neri;

²Questo esercizio ha da ritenersi facoltativo in quanto può risultare laborioso, benchè non difficile.

- T e t rappresentano (casella occupata da) una torre, C e c rappresentano un cavallo, A e a rappresentano un alfiere, D e d rappresentano una donna, R e r rappresentano un re.

Ricordiamo che un re è *sotto scacco* se esiste un pezzo avversario che attacca la casella in cui si trova. Dovete scrivere un programma C che legge: una posizione del gioco del gioco degli scacchi³ risponde:

- 0 se nessuno dei due re è sotto scacco;
- un intero positivo k se ci sono k pezzi bianchi che attaccano il re nero;
- un intero negativo $-k$ se ci sono k pezzi neri che attaccano il re bianco.

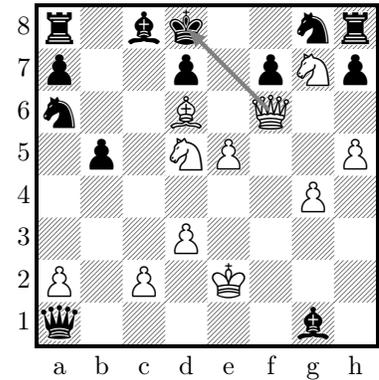
Supponete che i due re non siano mai contemporaneamente sotto scacco⁴.

Le due seguenti figure⁵ mostrano esempi di input e output del programma, e nel diagramma mostrano qual è la posizione corrispondente all'input ed esemplificano il motivo del numero tornato in output⁶.

Input:

t.ar.ct
p.p.pCp
c..A.D..
.p.CP..P
.....P.
...P....
P.P.R...
d....A.

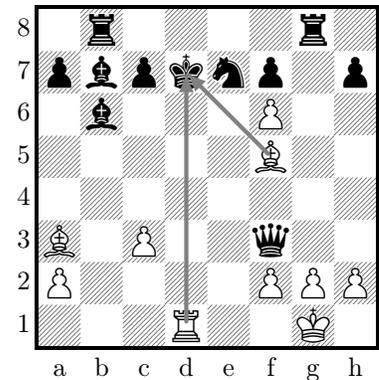
Output: 1



Input:

.t....t.
paprcp.p
.a...P..
.....A..
.....
A.P..d..
P....PPP
...T..R.

Output: 2



³potete usare sia `scanf("%c",...)`; e leggere un carattere alla volta, oppure `scanf("%s",...)`; e leggere una riga alla volta. Assicuratevi solo che tutto avvenga correttamente!

⁴questo invariante è garantito in posizioni legali del gioco degli scacchi!

⁵Un bonus di **un punto** agli scacchisti che indovineranno in quali partite sono apparse le due posizioni mostrate sopra ☺, scherzo ☺, ma vorrei tanto.

⁶le numerazioni di righe e colonne del diagramma in questo caso sono quelle tradizionali degli scacchi. Potete ignorarle e non confondetevi.