

Esercizio 1: Assembly

Si consideri una matrice di 4×4 locazioni, ciascuna contenente un numero naturale su 2 cifre decimali. Le coordinate di una locazione sono date da una lettera minuscola da a a d, che individua la colonna, e da una cifra da 1 a 4, che individua la riga. La locazione in alto a sinistra ha quindi coordinate a1, quella in basso a destra d4. Un esempio di contenuto della matrice, con le coordinate delle sue locazioni, è riportato in fig. 1.

	a	b	c	d
1	00	17	05	42
2	99	08	63	00
3	31	00	70	26
4	04	55	12	89

Figura 1: Esempio di matrice

Sono date le seguenti specifiche per i sottoprogrammi `read_coord`, `read_value` e `print_matrix`:

- `read_coord` legge da tastiera, facendone eco, le coordinate di una locazione: prima una lettera da a a d, poi una cifra da 1 a 4. La validazione è fatta carattere per carattere: durante ciascuna delle due letture il sottoprogramma ignora e non fa eco di ogni carattere inatteso, rimanendo in attesa di un carattere valido. Il sottoprogramma riporta, in un registro opportuno, un opportuno indice, offset o indirizzo per la locazione indicata dalle coordinate lette.
- `read_value` legge da tastiera, facendone eco, esattamente due cifre decimali, con la stessa validazione carattere per carattere descritta sopra. Lascia in un opportuno registro il numero naturale corrispondente.
- `print_matrix` riceve in un opportuno registro l'indirizzo della matrice e la stampa come descritto più avanti.

Sono date le seguenti specifiche per un programma che utilizzi i sottoprogrammi sopra descritti:

1. Inizializza a zero le 16 locazioni della matrice, e seleziona la locazione a1.
2. Iterativamente, legge un carattere di comando e lo esegue. I comandi possibili sono:
 - `m`: legge da tastiera le coordinate di una locazione, e la seleziona;
 - `i`: legge da tastiera un numero di due cifre decimali e lo scrive nella locazione attualmente selezionata;
 - `s`: stampa la matrice;
 - `q`: termina il programma.

Il carattere di comando è oggetto di una lettura a sé stante: il programma lo preleva immediatamente e, se è un comando valido, ne fa eco per segnalare di essere passato nella modalità corrispondente. L'eventuale argomento del comando è oggetto di letture successive, delegate al relativo sottoprogramma. I caratteri che non corrispondono ad alcun comando vengono ignorati, come per le altre letture.

Le stampe sono formattate come segue:

- Dopo l'eco del carattere di comando, i comandi `m` ed `i` stampano uno spazio, poi l'eco dell'argomento letto, e infine vanno a capo.
- I comandi `s` e `q` vanno a capo subito dopo l'eco del carattere di comando.
- La stampa della matrice occupa 4 righe, dalla riga 1 alla riga 4. Ciascuna riga contiene i 4 valori della riga corrispondente, ciascuno su 2 cifre decimali, separati da uno spazio.

Esempio di esecuzione del programma:

```
m b2
i 52
m d4
i 07
s
00 00 00 00
00 52 00 00
00 00 00 00
00 00 00 07
q
```

Le stampe vanno formattate come dagli esempi allegati al link: <https://tinyurl.com/47nbtaje>

Si ponga attenzione alla formattazione di questi file, che fa parte delle specifiche.

La documentazione Assembler in formato PDF è scaricabile al link: <https://tinyurl.com/mryny9jt>