

## Esercizio 1: Assembler

Scrivere un programma Assembler che si comporta come segue:

1. Legge e fa eco di esattamente 12 caratteri ASCII, costituiti da lettere minuscole, maiuscole o spazi. Non considera e non fa eco di caratteri inattesi.
2. Rimane ciclicamente in attesa di un comando, rappresentato da uno dei seguenti caratteri. Nota: non fa eco di alcun carattere di comando, né considera caratteri non associati a comandi.
  - c** : Trasforma, nella riga letta, ciascuna lettera minuscola nel corrispettivo maiuscolo. Stampa quindi la riga trasformata.
  - l** ("elle"): Trasforma, nella riga letta, ciascuna lettera maiuscola nel corrispettivo minuscolo. Stampa quindi la riga trasformata.
  - r** : Inverte l'ordine dei caratteri nella riga. Stampa quindi la riga trasformata.
  - n** : Legge una nuova riga, sovrascrivendo la precedente. Segue lo stesso comportamento del punto 1.
  - f** : Stampa una riga bianca e termina.

Per una valutazione positiva è sufficiente, alla consegna, l'implementazione di *uno* tra i comandi c e l. Fa parte della valutazione l'uso corretto e opportuno di sottoprogrammi e istruzioni stringa.

### Esempio

Assumendo che la sequenza di caratteri battuti sulla tastiera sia

```
poche parole  
cblzr!?c lf
```

Sullo schermo dovrà comparire la seguente sequenza di stringhe:

```
poche parole  
POCHE PAROLE  
poche parole  
elorap ehcop  
ELORAP EHCOP  
elorap ehcop
```

L'esempio di output è allegato anche in formato .txt al link: <http://tinyurl.com/mw66rkr4>. Si ponga attenzione alla formattazione di questo file, che fa parte delle specifiche.