

Laboratorio 1



R. Ferrero

Politecnico di Torino

Dipartimento di Automatica e Informatica (DAUIN)

Torino - Italy

This work is licensed under the Creative Commons (CC BY-SA) License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/>



Esercizio

Si scriva un programma in Assembly 8086 in grado di:

1. leggere in input un breve testo di 4 righe, ciascuna con un minimo di 20 caratteri ed un massimo di 50 caratteri.
2. contare il numero di occorrenze delle lettere.
3. applicare un algoritmo crittografico.

Lettura

- Il programma legge le righe con l'istruzione `INT 21H` e le memorizza nei vettori
`prima_riga DB 50 DUP(?)`
`seconda_riga DB 50 DUP(?)`
`terza_riga DB 50 DUP(?)`
`quarta_riga DB 50 DUP(?)`

Termine della lettura

- La lettura di una riga termina al verificarsi di una delle condizioni seguenti:
 - dopo almeno 20 caratteri, è stato letto INVIO
 - sono stati letti 50 caratteri senza nessun INVIO dopo i primi 20 caratteri
- Il carattere di INVIO corrisponde all'elemento 13 nella tabella ASCII
- Si noti che se si legge un INVIO nei primi 20 caratteri, la lettura della riga deve continuare.

Numero di occorrenze

- Per ogni riga, il programma deve contare quante volte ciascun carattere compare.
- Si considerino solo i caratteri $a...z$, $A...Z$, distinguendo tra maiuscole e minuscole.
- Per ogni riga, si individui il carattere più frequente (che compare MAX volte).
- Per ogni riga, si stampi l'elenco dei caratteri che compaiono almeno $MAX/2$ volte.
- Dopo ciascun carattere stampato, si scriva il numero di occorrenze.

Algoritmo crittografico

- Si stampi il testo usando il cifrario di Cesare, applicato solamente ai caratteri $a...z$, $A...Z$.
- Dato il parametro k , il cifrario di Cesare trasforma la lettera a in $a + k$, considerando la successione $a...zA...Za...zA...Z$ ecc.
- Caratteri non alfabetici rimangono invariati.
- $k = 1$ per la prima riga, 2 per la seconda, 3 per la terza, 4 per la quarta.
- Esempio con $k = 3$: piZza -> slcCd