

# Stringhe

## 1

Scrivere un programma che conti quante vocali sono presenti in una stringa inserita dall'utente. Utilizzare le funzioni di manipolazione delle stringhe (ad esempio `strlen(char*)` per la lunghezza della stringa).

## 2

Scrivere un programma che inverta una stringa inserita dall'utente.

Bonus: la stringa deve essere passata come argomento da linea di comando, ad esempio

```
$ ./inverti "stringa da invertire"  
eritrevni ad agnirts
```

## 3

Scrivere un programma che conti quante parole sono presenti in una stringa passata come argomento. Ad esempio

```
$ ./conta "word1 word2 3 word3"  
4
```

## 4

Scrivere un programma che permetta all'utente di sostituire tutte le occorrenze di una sottostringa in un testo con un'altra stringa. L'utente dovrà inserire

- il testo
- la stringa da cercare
- la stringa nuova da inserire

La stringa da cercare si intende formata da caratteri contigui nel testo in cui la si cerca. Inoltre è necessario assicurarsi che la stringa nuova sia lunga tanto quanto la sottostringa cercata; se questo non accade, segnalarlo opportunamente all'utente e chiedere di immettere una nuova stringa, indicando sia la stringa che si sta cercando sia la sua lunghezza.

## I/O da terminale

### 5

Scrivere un programma che funga da calcolatrice da terminale, consentendo all'utente di specificare operazioni matematiche e operandi come argomenti da riga di comando.

Ad esempio

```
$ ./calcolatrice 5 + 5  
10
```

```
$ ./calcolatrice 2 * 3  
6
```

```
$ ./calcolatrice 3 / 0  
err: divisione per zero
```

```
$ ./calcolatrice 3 / 2  
1.50
```

## I/O da file

### 6

Scrivere un programma che stampi a video il numero di occorrenze di ciascun carattere presente in un file di testo chiamato `input.txt`.

*Note*

- È possibile leggere il file di testo una sola volta!
- Controllare che il file esista e terminare il programma con un messaggio d'errore in caso contrario (usare `exit()` e `cerr`)

*Hint*

- Quanti elementi ha la tabella ASCII?
- Una variabile di tipo `char` può essere castata a ...

## 7

Scrivere un programma che prenda in ingresso due file di testo (`input.txt` e `output.txt`). Il file `output.txt` dovrà contenere le stesse parole del file `input.txt`, ma dovranno essere *censurate* sostituendo tutte le lettere in maiuscolo con ‘\*’. La procedura di censura di una stringa dovrà essere ricorsiva.

### *Note*

- È possibile usare la funzione `strlen` e `isupper`, incluse in `cstring`
- Fare tutti i controlli del caso
  - Numero di argomenti in input
  - Esistenza del file di testo