

# Laboratorio di Elementi di Architetture e Sistemi Operativi

Esercizi del 18 Aprile 2012

**Esercizio 1.** Modificare l'Esercizio 3 della lezione scorsa (occorrenze di ogni lettera dell'alfabeto) in modo che legga l'input da un file anziché dallo standard input. Il nome del file viene passato come primo argomento del main dalla linea di comando. Si assuma per semplicità che il nome del file non contenga spazi, e si gestiscano correttamente gli errori.

## Esercizio 2.

1. Scrivere una funzione con il seguente prototipo:  
`int strindex(char *s, char *t);`  
che restituisce la posizione (indice del vettore) di inizio della prima occorrenza della stringa `t` in `s`, oppure `-1` se `t` non compare in `s`.
2. Scrivere una versione semplificata del comando `grep` che prenda come argomenti una stringa e due nomi di file, e scriva tutte le linee del primo file contenenti almeno un'occorrenza della stringa nel secondo file. Si assuma per semplicità che la stringa ed i nomi dei file non contengano spazi, che le righe siano lunghe al massimo 80 caratteri, e si gestiscano correttamente gli errori.

## Esercizio 3.

1. Realizzare un insieme di funzioni per gestire una pila (stack) di valori floating point. In particolare, si implementino le operazioni di inserimento (`pop`) ed estrazione (`push`) di un valore nella pila. Si assuma che la pila possa contenere al massimo 10 valori.
2. Utilizzare le funzioni di gestione della pila per implementare una calcolatrice che esegua le quattro operazioni aritmetiche (+, -, \*, /) usando la notazione polacca inversa (RPN). La calcolatrice riceve le operazioni da eseguire dalla tastiera, leggendo un operatore o operando per ogni riga. Dopo aver letto un numero o un operatore (ed aver eseguito l'operazione) mostra i valori contenuti nello stack. Il programma termina quando l'utente inserisce `q`.

## Esempio:

```
$ ./calcolatrice
Calcolatrice RPN
2
Stack: 2
3
Stack: 2 3
-
Stack: -1
4
Stack: -1 4
5
Stack: -1 4 5
+
Stack: -1 9
/
Stack: -0.111111
a
ERRORE! Il dato inserito non è un numero né un operatore.
4
Stack: -0.111111 4
+
Stack: 3.888889
q
```