

## Esercizi per il sesto laboratorio

Antonio Miele      Ettore Speciale      Michele Tartara

2 novembre 2010

4 novembre 2010

5 novembre 2010

### Rubrica persistente - paddressbook

Scrivere un programma che gestisce una rubrica. Essa è composta da un insieme di elementi, ognuno dei quali rappresenta una persona (vedi Tabella 1). La rubrica contiene, al massimo, i dati di 30 persone e permette di effettuare le seguenti operazioni (a scelta dell'utente):

- inserire i dati di una nuova persona;
- salvataggio della rubrica su un file di testo;
- caricamento della rubrica da un file di testo;
- visualizzazione del contenuto della rubrica.

Il programma presenta le scelte di cui sopra tramite un menù. Selezionando un'opzione, l'utente avvia l'esecuzione dell'operazione scelta. Al termine dell'operazione, deve essere stampato un messaggio riassuntivo, seguito dal menù iniziale, in modo che l'utente possa selezionare una nuova operazione.

### Note

Alcune informazioni generali:

- nel menù, ovviamente, deve essere presente una voce per poter terminare il programma;
- è naturale rappresentare una persona tramite un tipo di dato strutturato con le caratteristiche elencate in Tabella 1;
- si assume che nè il nome nè il cognome contengano spazi;
- per leggere da file si può usare la funzione `fscanf`;
- per risolvere l'esercizio usate solo i costrutti che avete visto a lezione.

Tabella 1: caratteristiche di una persona

Campo	Tipo di Dato	Note
nome	stringa	massimo 16 caratteri, non contiene spazi
cognome	stringa	massimo 16 caratteri, non contiene spazi
codice fiscale	stringa	esattamente 16 caratteri
età	intero	positivo
sex	carattere	solo due valori permessi

## Estensione facoltativa

Si estenda la rubrica in modo che supporti anche le seguenti operazioni:

- calcolo della media dell'età delle persone;
- conteggio delle persone di sesso maschile e di quelle di sesso femminile.

## Invertitore di testo - inverter

Scrivere un programma che legge da un file di testo una sequenza di parole. Per ogni parola  $p$  ne calcola l'inversa  $p^R$ , la quale viene salvata su un file di output.

## Esempio

Contenuto del file letto:

*ma che bella giornata*

Contenuto del file generato dal programma:

*am ehc alleb atanroig*

## Note

Alcune informazioni generali:

- il file di input contiene parole separate da uno spazio o da un carattere di ritorno a capo (`\n`), quindi consigliamo di usare `fscanf` con lo specificatore di formato `%s` per leggere stringhe separate da spazi;
- le parole sono costituite da massimo 50 caratteri;
- le parole nel file di output devono essere separate da uno spazio.

## Decifratore - decoder

Il file allegato, `message.dat`, è un file binario contenente una sequenza di numeri interi. Scrivere un programma che:

- legge i numeri dal file binario, usando la funzione `fread`;
- li scrive in formato testuale su un nuovo file, separandoli con degli spazi.