

Prova scritta del 18/2/2011

Strutturare adeguatamente i programmi ed evidenziarne la strutturazione mediante indentazione. Inserire anche adeguati commenti

1) Realizzare una funzione di nome `aggiungi_suffisso` che, presi come suoi parametri due stringhe `A` e `B`, copia `A` in `B` aggiungendo il suffisso " - REV" prima dell'ultimo punto eventualmente presente in `A`. Se `A` non contiene punti, il suffisso è aggiunto in fondo.

Esempi

`A = risultati.11.1.2011.txt` `B = risultati.11.1.2011 - REV.txt`
oppure

`A = prova` `B = prova - REV`

Note. La stringa `A` non deve essere modificata. `A` e `B` devono essere stringhe "tipo C".

2) (a) Realizzare una funzione di nome `arrotonda` che preso come suo parametro un numero reale `x`, con parte intera `i` e parte decimale `d`, restituisce come suo risultato: `i` se $d \leq 0.3$; $i+0.5$ se $0.3 < d < 0.7$; $i+1$ se $d \geq 0.7$ (ved. es. al punto (b)). N.B. Si ricordi che assegnare un `float` ad una variabile `int` ne comporta il troncamento (= solo parte intera).

(b) Scrivere un programma principale che: legge da un file una sequenza di dati, separati da "a capo", ciascuno dei quali è costituito da un *cognome* e da tre *voti* numerici (numeri reali); per ciascun dato letto, viene calcolata la *somma* dei tre voti ed il risultato viene arrotondato tramite la funzione `arrotonda`; per ciascun dato così elaborato, il *cognome* ed il *risultato* arrotondato vengono scritti su un nuovo file (seguiti da "a capo"). Il nome del file di input è specificato (tramite `std input`) dall'utente, mentre il nome del file di output è costruito dal programma richiamando la funzione `aggiungi_suffisso` sul nome del file di input.

Esempio

File voti.txt

```
rossi 8.5 9.75 7.0
bianchi 8.7 9.75 7.0
verdi 8.5 9.75 7.7
```

File voti - REV.txt

```
rossi 25
bianchi 25.5
verdi 26
```

Note. Non memorizzare i dati letti dal file di input in un array.

3) Sia `elem` un tipo `struct` costituito da un campo `nome` di tipo *stringa* (max. 50 caratteri) e da tre campi, `x_1`, `x_2`, `x_3`, di tipo *intero*. Realizzare una funzione di nome `media` che, presi come suoi parametri un array `A` di elementi di tipo `elem`, il numero `n` di elementi in `A` e un intero `i` ($1 \leq i \leq 3$), calcola e restituisce come suo risultato la media dei campi `x_i` di tutti gli elementi di `A`.

Esempio.

Se `A = {{"alfa", 3, 4, 5}, {"beta", 5, 6, 7}, {"gamma", 7, 8, 9}}` ($n = 3$)
e $i = 2$, la media dei campi `x_2` è 9.