

Prova di laboratorio di Fondamenti di Informatica

Ettore Speziale

20 gennaio 2011

ISTRUZIONI:

- il tempo a disposizione è 1:30;
- è possibile consultare qualsiasi libro e i propri appunti;
- è vietato copiare, parlare o consultarsi con il vicino pena annullamento della prova;
- specificare all'inizio di ciascun file sorgente il proprio nome, cognome e matricola in un commento;
- i file devono avere come nome il numero dell'esercizio preceduto dal prefisso `es` (`es1.c`, `es2.c`, `es3.c`);
- i file sorgente devono essere salvati in una cartella dal nome COGNOME_MATRICOLO (es: Mario Bianchi, matr. 111111, chiamerà la propria cartella Bianchi_111111).

Esercizio 1

Scrivere un programma che chiede all'utente di inserire una stringa s di massimo 50 caratteri ed un numero naturale n minore della lunghezza effettiva della stringa. Il programma crea una nuova stringa s' , di lunghezza pari a quella di s , in cui i primi n caratteri sono *. I restanti sono i primi caratteri della stringa s (esempio: $s = abcde, n = 2 \rightarrow **abc$). Successivamente la stringa s' viene stampata a video.

Esercizio 2

Nel file `lista.c` è implementato un programma che utilizza una lista concatenata per memorizzare una serie di numeri naturali di lunghezza indefinita. Il programma chiede all'utente una serie di numeri da memorizzare nella lista ed effettua l'inserimento mediante il sottoprogramma `inserisci()` (inserimento in coda); la fase di inserimento termina quando l'utente inserisce un numero negativo. In seguito il programma stampa a video il contenuto della lista mediante il sottoprogramma `visualizza()`.

Estendere il programma fornito nel seguente modo:

- scrivere un sottoprogramma `sommaCoppie()` che riceve in ingresso una lista l di numeri naturali; il sottoprogramma scorre la lista l e per ogni coppia aggiunge ad una nuova lista l' un nuovo numero uguale alla somma dei due elementi della coppia. Nel caso la dimensione della lista l sia dispari, l'ultimo elemento viene ricopiato nella lista l' . Esempio:

$$8\ 5\ 16\ 7\ 3 \rightarrow \underbrace{8\ 5}\ \underbrace{16\ 7}\ \underbrace{3} \rightarrow 13\ 23\ 3$$

- modificare il `main()` del programma fornito aggiungendo una chiamata al sottoprogramma implementato dopo la visualizzazione dei dati inseriti, e visualizzando la nuova lista.

Esercizio 3

Scrivere un programma che chiede all'utente una parola, *originale* di massimo 20 caratteri. Il programma apre il file di testo `testo.dat` contenente una serie di parole lunghe al massimo 20 caratteri. Il programma crea un nuovo file `result.dat`, ricopiando le parole lette dal primo; nel caso la parola corrente sia uguale ad *originale* la ricopia due volte.

Esempio

originale = ciao

Contenuto del file `testo.dat`:

ciao mondo a vercana c'è il sole ciao ciao anche a buccinasco

Contenuto del file `result.dat`:

ciao ciao mondo a vercana c'è il sole ciao ciao ciao ciao
anche a buccinasco