

Prova di laboratorio di Fondamenti di Informatica

Antonio Miele

18 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 di massimo 50 caratteri. Il programma crea la stringa inversa (esempio: `abcde` \rightarrow `edcba`), memorizzandola in un nuovo vettore e la stampa 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 `eliminaMultipliDiN()` che riceve in ingresso una lista `l` di numeri naturali ed un intero `n`; il sottoprogramma crea una nuova lista contenente tutti i numeri non multipli di `n` della lista `l`. Esempio:

$n = 4$
8 5 16 7 \rightarrow 5 7

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

Esercizio 3

Scrivere un programma che chiede all'utente due parole, *originale* e *sostituta*, 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 sostituisce con la parola *sostituta*.

Esempio

originale = `ciao`
sostituta = `hello`

Contenuto del file `testo.dat`:

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

Contenuto del file `result.dat`:

```
hello mondo a vercana c'è il sole  
hello hello anche a buccinasco
```