

Fondamenti di Informatica T

Corso di Laurea Triennale in Ingegneria Elettrica – a. a. 2010/2011

Prova d'esame 8 Giugno 2011 – Docenti: prof. Rebecca Montanari, dott. Michele Lombardi

Si progetti un programma che consente di consultare un catalogo di voli (tutti per una specifica data). La lista dei voli è memorizzata in un file di testo "voli.txt" (allegato al testo dell'esame).

Ogni volo occupa 5 righe nel file, in particolare:

- la prima riga contiene il codice del volo (stringa di al più 10 caratteri, no spazi)
- la seconda riga contiene la città di partenza (stringa di al più 30 caratteri, no spazi)
- la terza riga contiene la città di arrivo (stringa di al più 30 caratteri, no spazi)
- la quarta riga contiene l'orario di partenza (ore e poi minuti, interi separati da uno spazio)
- la quinta riga contiene l'orario di arrivo (ore e poi minuti, interi separati da uno spazio)

Si assuma che il file non possa contenere più di 40 voli.

Quesito 1:

Si scriva il codice della funzione:

```
int leggi_voli(char* file_voli, Volo v[]);
```

che legga da un file specificato dall'utente ("file_voli") i dati su una serie di voli e li inserisca nell'array "v" di strutture Volo (da definire opportunamente). La funzione restituisce il numero di voli contenuti nel file. Nel caso non sia possibile aprire il file, la funzione fa terminare il programma con un messaggio di errore.

Quesito 2:

Si scriva un semplice programma che utilizza la funzione "leggi_voli" per caricare il catalogo dal file "voli.txt"; si visualizza o a video la lista dei voli.

Quesito 3:

Si scriva il codice della funzione:

```
int filtra_partenza(Volo v[], int dim, Volo res[], char* partenza);
```

che esamina un array di voli "v" di dimensione "dim" ed inserisce in "res" i voli che partono da una città specificata dall'utente ("partenza"). La funzione restituisce il numero di elementi in "partenza".

Quesito 4:

Si scriva il codice della funzione:

```
int filtra_arrivo(Volo v[], int dim, Volo res[], char* arrivo);
```

del tutto analoga alla precedente, ma che permette all'utente di specificare una città di arrivo.

Quesito 5:

Si estenda il programma, richiedendo all'utente una città di partenza ed una di arrivo e visualizzando tutti i voli che connettono direttamente le due città.

Quesito 6:

Si scriva il codice della funzione:

```
int combina_voli(Volo a, Volo b);
```

che restituisce 1 se i voli "a" e "b" possono essere effettuati in sequenza e 0 altrimenti. I voli si possono combinare se:

- la città di arrivo di "a" è la città di partenza di "b"
- l'orario di arrivo di "a" precede l'orario di partenza di "b"

Quesito 7:

Si estenda il programma in modo che visualizzi tutte le connessioni con uno scalo tra le città di partenza e di arrivo specificate dall'utente.