

Fondamenti di Informatica L-A (A.A. 2003/2004) - Ingegneria Informatica
Prof.ssa Mello & Prof. Bellavista – Prova d'Esame di Lunedì 12/01/2004 – durata 2h:30m
COMPITO B

ESERCIZIO 1 (12 punti)

Sono dati due file di testo **opere.txt** ed **esecuzioni.txt** che contengono, rispettivamente, i dati relativi ad alcune opere liriche e l'elenco delle esecuzioni disponibili in un negozio di dischi. Più precisamente, ogni riga di **opere.txt** contiene, nell'ordine:

- Codice opera (numero intero), uno e un solo spazio di separazione;
- Nome dell'opera (non più di 40 caratteri senza spazi), uno e un solo spazio di separazione;
- Nome del compositore (non più di 30 caratteri senza spazi), terminatore di riga.

Ogni opera compare nel file **opere.txt** una ed una sola volta.

Ogni riga di **esecuzioni.txt** contiene, nell'ordine:

- Codice opera (numero intero), uno e un solo spazio di separazione;
- Nome del direttore d'orchestra (non più di 30 caratteri senza spazi), uno e un solo spazio di separazione;
- Costo del disco (numero reale), uno e un solo spazio di separazione;
- Un carattere ('d' se l'esecuzione è registrata in digitale, 'a' se analogica), terminatore di riga.

Ad esempio:

| opere.txt | esecuzioni.txt |
|-------------------|-----------------------|
| 1 Traviata Verdi | 1 Muti 36.00 d |
| 2 Rigoletto Verdi | 1 Toscanini 39.00 a |
| 3 Requiem Mozart | 3 Karajan 19.00 a |
| | 1 Abbado 28.90 a |
| | |

Si scriva una procedura **load()** che riceva come parametri di ingresso due puntatori a file di testo e restituisca come parametri di uscita un vettore **y** contenente strutture **disco** (nome opera, costo) e il numero degli elementi **N** inseriti in **y**: questo vettore deve contenere solo i dati relativi alle esecuzioni registrate in analogico. Si ricorda inoltre l'esistenza della procedura di libreria **void rewind (FILE *f)** che riporta la testina di lettura a inizio file (**6 punti**).

Si scriva un programma C che, utilizzando la procedura **load()** precedentemente definita, inserisca in un vettore **disponibili** (supposto di dimensione massima **DIM=100**) le strutture **disco** di cui sopra, derivanti dai file **opere.txt** ed **esecuzioni.txt**. Il programma chieda poi all'utente il nome di un'opera, e stampi il costo medio di tutte le versioni analogiche di tale opera (**6 punti**).

ESERCIZIO 2 (7 punti)

Si scriva una funzione **d()** che data in ingresso una lista ordinata di interi **l1**, restituisca in uscita una nuova lista, ottenuta da **l1** eliminando tutti gli elementi ripetuti. Ad esempio, se invocata con **l1=[-5, 0, 0, 0, 1, 7, 7]**, la funzione **d()** deve restituire la lista **[-5, 0, 1, 7]**.

La funzione **d()** può essere realizzata in modo ricorsivo o iterativo, utilizzando solo ed esclusivamente la **rappresentazione collegata a puntatori** e **NON** le operazioni primitive definite durante il corso.

ESERCIZIO 3 (6 punti)

Il seguente programma C compila correttamente? In caso affermativo, quali sono i valori stampati a tempo di esecuzione (si motivi opportunamente la risposta data)?

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define DIM 9

int Funz(char *, int *, int *);
int N = DIM - 5;

main () {
    int i; char ss[DIM];
    char * s = (char *) malloc(DIM);
    strcpy(s, "Paperino"); i=1;
    N = (--N)+2;
    N = Funz(s, &N, &i);
    printf("N vale adesso: %d\n",N);
    {
        for (i=0; i<DIM-1; ++i) {
            ss[i] = *(s+i);
            printf("%c", *(ss+i)); }
    }
}

int Funz(char * y, int *N, int *M) {
    int i;
    (*N)++;
    for (i=(*N)+1; i>(*N); i--)    y[i] = y[*M];
    return *N;
}
```

Esercizio 4 (6 punti)

Si consideri la seguente funzione G:

```
double G(float x){
    x=x+1;
    if (x<0) return G(x)+G(x+2);
    else return 1;}

```

Si scriva il risultato della funzione quando invocata come G(-3) e si disegnino i corrispondenti record di attivazione.

Esercizio 5 (2 punti)

Un elaboratore rappresenta i numeri interi su 8 bit dei quali 7 sono dedicati alla rappresentazione del modulo del numero e uno al suo segno. Indicare come viene svolta la seguente operazione aritmetica e determinarne il risultato trasladolo poi in decimale per la verifica:

SOLUZIONE COMPITO B

ESERCIZIO 1

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define DIM 100

typedef struct {
    char nome[41];
    float costo;
} disco;

void load(FILE * f1, FILE * f2, disco * y, int * num) {
    int codicel, codice2;
    char nome[41], autore[31], artista[31], tipo;
    float costo;

    *num=0;
    while ( fscanf(f1,"%d %s %s\n", &codicel, nome, autore) != EOF )
        { while ( fscanf(f2,"%d %s %f %c\n", &codice2, artista, &costo, &tipo)
            != EOF )
            if ( (codicel==codice2) && (tipo=='a') ) {
                strcpy(y[*num].nome, nome);
                y[*num].costo = costo;
                (*num)++;
            }
        }
    rewind(f2);
}

main() {
    FILE *f1, *f2;
    int N, i, count;
    float somma;
    char nome[41];
    disco disponibili[DIM];

    if ((f1=fopen("opere.txt", "r"))==NULL) {
        printf("Non sono riuscito ad aprire file1 in lettura!"); exit(1); }
    if ((f2=fopen("esecuzioni.txt", "r"))==NULL) {
        printf("Non sono riuscito ad aprire file2 in lettura!"); exit(1); }

    load(f1, f2, disponibili, &N);
    fclose(f1); fclose(f2);

    printf("Inserire nome opera: ");
    scanf("%s", nome);

    somma = 0; count = 0;
    for (i=0; i<N; i++)
        if (!strcmp(nome, disponibili[i].nome))
            { somma = somma + disponibili[i].costo; count++; }
    printf("Costo medio per l'opera %s: %f\n", nome, somma/count);
}
```

ESERCIZIO 2

```
list d(list l1) {
    if (l1==NULL) return l1;
    else if (l1->next == null) return l1;
    else {
        if (l1->value == (l1->next)->value) return d(l1->next);
        else { l1->next = d(l1->next);
              return l1; }
    }
}
```

ESERCIZIO 3

Il programma è corretto sintatticamente e stampa:

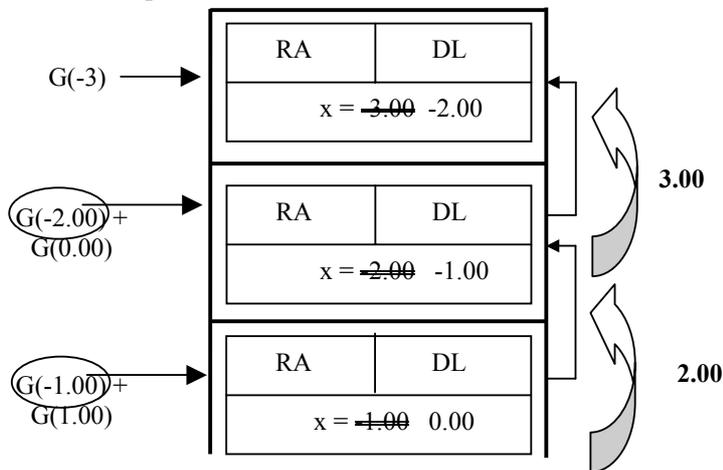
N vale adesso: 6

Paperina

Infatti la stringa s in seguito alla chiamata di strcpy() viene inizializzata a "Paperino", terminatore compreso. Successivamente viene effettuato un semplice calcolo su N, dichiarato globalmente. Prima dell'invocazione di Funz(), N vale 5. La funzione Funz() è quindi invocata con parametri il puntatore a caratteri s, e valori 5 e 1. La funzione Funz() si limita a sostituire alcuni caratteri in determinate posizioni; in particolare il ciclo sostituisce solo il carattere in posizione 7 con il carattere in posizione 1. Il valore restituito è il secondo parametro incrementato di 1, ovvero 6. Infine il programma copia nell'array di caratteri ss la stringa puntata da s, e di volta in volta stampa anche tali caratteri.

ESERCIZIO 4

La funzione restituisce il valore 3. Supponendo che la valutazione degli addendi nella somma venga fatta a partire da sinistra, si ottiene prima:



poi l'eliminazione dell'ultimo record, e un nuovo ulteriore record di attivazione per G(1.00). Quindi viene fatta la prima somma con risultato 2, restituita indietro dove viene fatta l'ultima invocazione di G(0.00), con somma finale restituita alla prima invocazione di G().

ESERCIZIO 5

46-> 0 0101110

93-> 0 1011101

Tra i moduli dei numeri si esegue una sottrazione ottenendo: 1 0101111

che vale -47 in base dieci.