

Fondamenti di Informatica L-A (A.A. 2005/2006) - Ingegneria Informatica
Prof.ssa Mello & Prof. Bellavista – Prova d’Esame del 22/06/2006 - durata 2h30m
COMPITO A

ESERCIZIO 1 (13 punti)

listino.txt
acqua fiuggi 7.0
acqua recoaro 6.0
pasta barilla 0.3
pasta dececco 0.5

Un negoziante tiene traccia del prezzo degli articoli in vendita e dell’elenco degli articoli già venduti in due file di testo distinti. In particolare, il file **listino.txt** specifica in ogni riga, separati tra loro da uno spazio, la tipologia di articolo in vendita (al più dieci caratteri senza spazi), la sua marca (al più 10 caratteri senza spazi) e il suo prezzo in euro (float). Il file **venduti.txt** elenca gli articoli già

venduti.txt
acqua recoaro
acqua recoaro
pasta barilla
pasta barilla
acqua recoaro
pasta dececco

venduti, con una riga con tipologia e marca per ogni articolo venduto. Ovviamente possono esserci più occorrenze di uno stesso articolo in **venduti.txt** e non è detto che ogni articolo presente in **listino.txt** sia presente anche in **venduti.txt**; invece se un articolo è presente in **venduti.txt** allora è sicuramente presente anche in **listino.txt**.

Dopo aver realizzato una struttura dati **item** in cui sia possibile specificare la tipologia di un articolo, la sua marca, il prezzo in euro e la quantità di articoli già venduti tramite un intero, il candidato realizzi:

1. una funzione

```
item* articoli(FILE* listino, FILE* venduti, char* tipologia, int* len)
```

che, ricevuti in ingresso due puntatori a file ed una tipologia di articolo (ad esempio **pasta**), legga il file **listino** per calcolare quanti articoli sono presenti del tipo **tipologia** e sfrutti tale valore per allocare dinamicamente memoria sufficiente per contenere tutti gli articoli di quel tipo presenti nel file **listino.txt**. Per ogni marca in vendita della tipologia di articolo richiesta, la funzione inserisca nello spazio di memoria allocata dinamicamente un **item**. Per ogni **item** specificare, oltre a tipologia, marca e prezzo, anche il numero di articoli venduti (ovvero il numero di occorrenze in **venduti**).

La funzione **articoli(...)** restituisca alla funzione chiamante un puntatore all’area di memoria che contiene gli **item** e il numero di elementi restituiti tramite **len**.

2. un programma **main()** che, aperti i file **listino.txt** e **venduti.txt**, richieda all’utente la tipologia dell’articolo di interesse, richiami opportunamente la funzione **articoli(...)**, e stampi a video tali **item** (tipologia, marca, prezzo e numero di articoli venduti).

Ad esempio, se l’utente inserisce “**pasta**”, la prima riga dell’output sarà: “**pasta barilla 0.3 2**” e la seconda “**pasta dececco 0.5 1**”.

Si ricorda l’esistenza della funzione **void rewind(*FILE)** che riporta la testina di lettura a inizio file e della funzione **int strcmp(char* st, char * ct)** per il confronto tra stringhe.

ESERCIZIO 2 (9 punti)

Si scriva una funzione **list compare(list l1, list l2)** che, ricevute in ingresso due liste di interi, restituisca una terza lista avente come elementi gli elementi di **l1** che compaiono in **l2** un numero di volte equivalente al valore dell’elemento di **l1** attualmente preso in considerazione. Ad esempio se **l1=[4,1,3,2,5]** e **l2=[1,4,2,6,3,4,4,1,2,4]**, gli elementi della lista restituita saranno **4 e 2** poiché tali elementi della lista **l1** compaiono rispettivamente quattro e due volte nella lista **l2**.

La funzione deve essere realizzata utilizzando il tipo di dato astratto **list**, definito per gli interi (non è necessario riportare la definizione nella soluzione). Si possono utilizzare le sole operazioni primitive definite durante il corso, che quindi possono NON essere riportate nella soluzione. Non si possono usare altre funzioni di alto livello.

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>

int inv(int val, char* type1, char* type2){
    int start=val, i, tot=0;
    int lim1=0,lim2=0;
    char ch;
    while(type1[lim1]!='\0') lim1++;
    while(type2[lim2]!='\0') lim2++;

    while(start<lim1 && start<lim2){
        for(i=start;i<lim1;i++){
            ch=*(type1+i);
            *(type1+i)=*(type2+lim2-i-1);
            *(type2+lim2-i-1)=ch;
            tot++;
        }
        start+=2;
    }
    return tot;
}

int main(){
    char *s1="abcde", *s2="12345678";
    int tot=inv(1,s1,s2);
    printf("%s %s %d\n", s1, s2, tot);
    return 0;
}
```

ESERCIZIO 4 (4 punti)

Data la funzione:

```
int fun(float f, int i){
    int mul=f*3;
    i*=2;
    if(i<mul) return fun(f/i, i)-1;
    else return i*f;
}
```

e la funzione chiamante:

```
int main(){
    int r=fun(12.8, 2);
    printf("%d\n",r);
    return 0;
}
```

mostrare la sequenza dei record di attivazione. Che cosa viene stampato sullo standard output?

Esercizio1

```
typedef struct{
    char tipo[11];
    char marca[11];
    float prezzo;
    int totaleVenduti;
}item;

item* articoli(FILE* listino, FILE* venduti, char* tipologia, int* len){
    char tipo[11], marca[11];
    item *res,*resTemp, temp;
    *len=0;
    while(fscanf(listino,"%s %s %f\n", temp.tipo, temp.marca, &(temp.prezzo))!=EOF){
        if(strcmp(tipologia,temp.tipo)==0){
            (*len)=(*len)+1;
        }
    }
    res=(item*)malloc( sizeof(item)*(*len) );
    resTemp=res;

    rewind(listino);
    while(fscanf(listino,"%s %s %f\n", temp.tipo, temp.marca, &(temp.prezzo))!=EOF){
        if(strcmp(tipologia,temp.tipo)==0){
            temp.totaleVenduti=0;
            while(fscanf(venduti,"%s %s\n", tipo, marca)!=EOF){
                if(strcmp(tipologia,tipo)==0){
                    if(strcmp(temp.marca,marca)==0){
                        (temp.totaleVenduti)++;
                    }
                }
            }
            rewind(venduti);
            (*resTemp)=temp;
            resTemp++;
        }
    }
    return res;
}

int main(){
    FILE *lis, *ven;
    char tipo[11];
    item *res;
    int numeroArt,i;
    if((lis=fopen("listino.txt","r"))==NULL){
        printf("Problemi durante l'apertura del file listino.txt\n");
        exit(-1); }
    if((ven=fopen("venduti.txt","r"))==NULL){
        printf("Problemi durante l'apertura del file venduti.txt\n");
        exit(-1); }

    printf("Tipologia?\n");
    scanf("%s",tipo);

    res=articoli(lis, ven, tipo, &numeroArt);
    close(lis); close(ven);

    for(i=0;i<numeroArt;i++){
        printf("%s %s %f %d\n", (res+i)->tipo, (res+i)->marca, (res+i)->prezzo,
(res+i)->totaleVenduti);
    }
    return 0;
}
```

Esercizio2

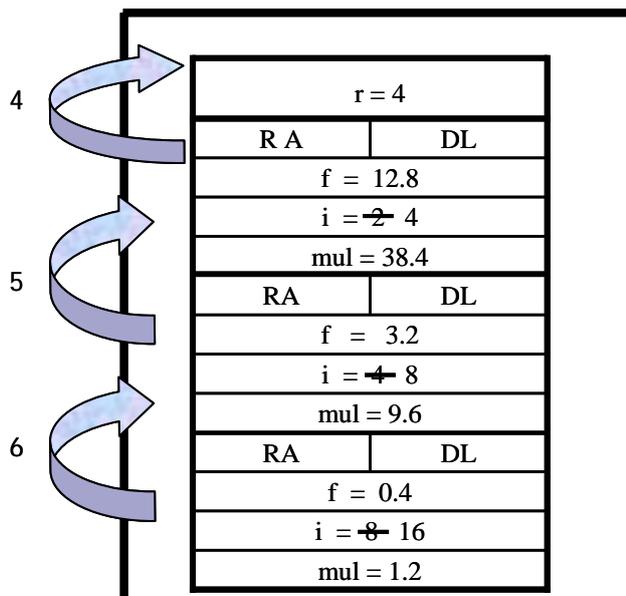
```
#include "list.h"
list compare(list l1, list l2){
    element el;
    int tot;
    list res=emptylist();
    list l2temp;
    while(!empty(l1)){
        tot=0;
        l2temp=l2;
        el=head(l1);
        while(!empty(l2temp)){
            if(head(l2temp)==el)tot++;
            l2temp=tail(l2temp);
        }
        if(el==tot)res=cons(el,res);
        l1=tail(l1);
    }
    return res;
}
```

Esercizio3

La funzione `inv(...)` prende in ingresso due stringhe ben formate e un int. Il ciclo `while` itera due volte: nel primo ciclo scambia i caratteri `bcde` di `type1` con i caratteri `7654` di `type2`, nel secondo ciclo scambia i caratteri `45` di `type1` con i caratteri `de` di `type2`. La funzione restituisce il numero totale di caratteri scambiati, ovvero 6.

`main()` richiama la funzione `inv(...)` e stampa sullo standard output `"a76de 12345cb8 6"`.

Esercizio 4



Sullo standard output viene scritto il risultato della chiamata alla funzione `fun(...)`, ovvero `"4"`.