



Cognome e nome: ..... Matricola: .....

### Quesito 1 (18 punti)

Punteggio ottenuto: .../18

Per gestire i prestiti di libri, una biblioteca utilizza 2 files:

- LIBRI.DAT che contiene i dati relativi ai libri; per ogni libro tale file riporta il titolo, il numero (naturale) di inventario e un campo intero (disponibile) che indica se il libro è presente in biblioteca (disponibile=1) o è in prestito (disponibile=0).
- UTENTI.DAT che riporta i dati delle persone che possono avere i libri in prestito. Per ciascuna persona sono specificati il nome, l'indirizzo ed il numero di inventario del libro attualmente in prestito (per semplicità supponiamo che ogni persona prenda in prestito al massimo un libro); se la persona non ha in prestito alcuno libro si assume abbia il libro con numero di inventario 0.

Si scriva un programma che:

1. A partire dal file LIBRI.DAT crei due liste, `presenti` e `in_prestito`, ordinate in base al numero di inventario, contenenti, rispettivamente, i dati dei libri presenti nella biblioteca e quelli dei libri usciti in prestito;
2. letto dallo standard input il numero di inventario di un libro, acceda alla lista `in_prestito` per determinare se esso è uscito e, in caso affermativo, accedendo al file UTENTI.DAT, stampi a video il nome della persona che lo ha in prestito;
3. stampi a video i contenuti delle due liste;
4. stampi a video i nomi di tutti quelli che hanno un libro in prestito.

esempio di LIBRI.DAT:

```
la_mano_nera 456 1
il_re_di_roma 321 0
la_rosa_scarlatta 987 0
il_piccolo_re 212 1
non_mi_guardare 444 0
la_grande_caccia 566 1
non_tornare 767 1
stasera_non_torno 989 0
```

esempio di UTENTI.DAT

```
andrea 566
franco 456
romano 0
giovanni 987
sandro 444
piera 566
anna 0
franca 0
filippo 767
```



Cognome e nome: ..... Matricola: .....

## Soluzione:

```
#include <iostream>
#include <stdlib.h>
#include <fstream>
#include <cstring>

using namespace std;

typedef struct elem {
    char titolo[20];
    int cod;
    int disp;
    elem* succ;
};

typedef elem* lista;

void leggi1(lista& p);
void leggi2(lista& p);
void inserimento(lista& p0, elem a);
void stampa(lista p0);
void cerca(int codice, char nome[]);

int main(int argc, char *argv[])
{
    lista presenti=0;
    lista in_prestito=0;
    int codice;
    char nome[15];

    leggi1(presenti);
    leggi2(in_prestito);
    cout <<"stampa dei libri presenti\n";
    stampa(presenti);
    cout <<"\nstampa dei libri in prestito\n";
    stampa(in_prestito);

    cout <<"dai il codice del libro\n";
    cin >> codice;
    cerca(codice,nome);
    cout <<"il libro e' in prestito a "<<nome<<endl;

    system("PAUSE");
    return 0;
}

void leggi1(lista& p)
{
    fstream input;
    elem a;

    input.open("libri.dat",ios::in);
```



Cognome e nome: ..... Matricola: .....

```

        while(input.good()) {
            input >>a.titolo >> a.cod>>a.disp;
            if(a.disp == 1)
                inserimento(p, a);
        }
        input.close();
    }

    void leggi2(lista& p)
    {
        fstream input;
        elem a;

        input.open("libri.dat",ios::in);

        while(input.good()) {
            input >>a.titolo >> a.cod>>a.disp;
            if(a.disp == 0)
                inserimento(p, a);
        }
        input.close();
    }

    void inserimento(lista& p0, elem a)
    {
        lista p = 0, q, r;
        for (q = p0; q != 0 && q->cod < a.cod; q = q->succ)
            p = q;
        r = new elem;
        strcpy(r->titolo, a.titolo);
        r->cod = a.cod;
        r->disp = a.disp;
        r->succ = q;
        // controlla se si deve inserire in testa
        if (q == p0)
            p0 = r;
        else
            p->succ = r;
    }

    void stampa(lista p0)
    {
        lista p = p0;
        while (p != 0)
        {
            cout << p->titolo << ' ' <<p->cod << ' ' << p->disp << endl;
            p = p->succ;
        }
        cout << '\n';
    }

    void cerca(int codice, char nome[])

```



Cognome e nome: ..... Matricola: .....

```
{
    fstream input;
    bool trovato = false;
    int cod;
    char nome1[15];

    input.open("utenti.dat", ios::in);

    while(input.good()) {
        input >> nome1 >> cod;
        if(codice == cod)
        {
            trovato = true;
            strcpy(nome, nome1);
            return;
        }
    }
    input.close();

    if(!trovato)
        strcpy(nome, "nessuno");
}
```