

# ELEMENTI DI INFORMATICA LB - PROVA PRATICA DEL 18/04/2007

## 1. LINGUAGGIO C (15 punti)

È dato un file di testo denominato LIBRI.TXT che contiene un elenco di informazioni relative ai libri presenti in una nota biblioteca bolognese. Il file contiene al massimo 20 righe e ogni riga contiene:

- un insieme di caratteri e cifre in cui è specificato un codice univoco che identifica il libro (8 caratteri alfanumerici)
- un codice alfanumerico (3 caratteri alfanumerici) che rappresenti le coordinate di dove si trova il libro (corridoio, scaffale, ripiano).
- un booleano che rappresenti la presenza o meno del libro sugli scaffali (nel caso sia stato prestato o meno).
- una data nel formato (aaaa/mm/gg) in cui il libro risulta essere stato prestato.

**Si chiede di scrivere un programma C, **Biblioteca**, con le seguenti caratteristiche:**

**Il programma dovrà essere suddiviso in 3 file.**

1. **funzioni.h:** dovrà contenere le definizioni dei tipi e le dichiarazioni delle funzioni sotto descritte
2. **funzioni.c:** dovrà contenere le definizioni delle funzioni dichiarate in funzioni.h
3. **main.c:** dovrà contenere solo la funzione main()

**Il programma dovrà comprendere:**

**Tipi di strutture**

1. **libro** destinato a contenere i dati di una riga del file LIBRI.TXT
2. **libri** che contiene un array di strutture di tipo **libro**, e il numero di righe effettivamente lette.

**Funzioni e procedure**

1. **funzione** denominata **leggi()**: dato il **nome di un file** riempie una lista di strutture di tipo **libro** contenente le righe lette dal file LIBRI.TXT. La funzione restituisce 1 se l'operazione è andata a buon fine, altrimenti restituisce 0.
2. **funzione** denominata **conta()**: data una **lista** di strutture di tipo **libro** restituisce il **numero di libri prestati prima del 1 aprile 2007**

**Il programma, utilizzando le suddette funzioni, dovrà:**

1. leggere il file **LIBRI.TXT** usando la funzione **leggi()** segnalando a video eventuali problemi;
2. usare la funzione **conta()** per calcolare il numero di libri prestati prima del 1 aprile 2007 e mostrarlo a video;

## **2. DATA BASE (16 punti)**

### **SPECIFICA**

Occorre gestire informazioni su :

- una casa editrice che produce libri aventi un codice ed un titolo.
- alcuni di essi sono libri di testo universitari, ed in tal caso sono relativi ad una specifica materia, e sono consigliati per uno o più Corsi Universitari, che vengono opportunamente indicati con nome del Corso di Laurea, e nome e anno del Corso stesso.
- altri non sono libri di testo, e per essi viene indicato l'argomento (saggistica, letteratura, manualistica, ecc.) e l'eventuale fascia di età alla quale sono indirizzati.
- ogni libro ha uno o più autori, ognuno dei quali ha un nome, un indirizzo, e, qualcuno, un nome d'arte.

### **SCHEMA LOGICO: CASA EDITRICE (9 punti)**

Progettare una base di dati che permetta di gestire il problema descritto sopra:

- definire uno schema Entità/Relazione che descriva il problema;
- definire uno schema logico secondo il modello relazionale, derivato da una ristrutturazione, se necessaria, del diagramma E/R.

### **SQL (7 punti)**

#### **1. CREARE LE TABELLE**

Si scrivano le istruzioni SQL necessarie a creare le tabelle dello schema logico sopra realizzato

#### **2. EFFETTUARE INTERROGAZIONI**

Scrivere l'istruzione SQL per la seguente query di selezione:

- elencare i titoli dei libri consigliati per i corsi universitari del 2° anno.

## **SOLUZIONI**

# 1. LINGUAGGIO C (15 punti)

## funzioni.h

```
typedef enum {false,true} boolean;

typedef struct
{
    char id[8+1];
    char coordinate[3+1];
    boolean prestato;
    char data[10+1];
} libro;

typedef struct list_element
{
    libro value;
    struct list_element *next;} item;

typedef item *list;

typedef struct
{
    libro p[19];
    int i;} libri;

/* ADT */
list emptyList(void);
boolean isEmpty(list);
libro head(list);
list tail(list);
list cons(libro,list);

/* Funzioni non Primitive */
void showList(list);
int leggi(char*,list*);
int conta(list);
```

## funzioni.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "funzioni.h"

list emptyList(void)
{
    return NULL;
}

boolean isEmpty(list l)
{
    if(l==NULL)
    {
        return true;
    }
    else
    {
```

```

        return false;
    }
}

libro head(list l)
{
    if(isEmpty(l))
    {
        abort();
    }
    else
    {
        return l->value;
    }
}

list tail(list l)
{
    if(isEmpty(l))
    {
        abort();
    }
    else
    {
        return l->next;
    }
}

list cons(libro e,list l)
{
    list t;
    t=(list) malloc(sizeof(item));
    t->value=e;
    t->next=l;
    return t;
}

int leggi(char* nomeFile,list *l)
{
    FILE *fp;
    libro p;
    int letti=0;

    list temp=emptyList();
    if((fp=fopen(nomeFile,"r"))==NULL)
    {
        return 0;
    }
    else
    {
        while(fscanf(fp,"%s%s%d%s\n",p.id,p.coordinate,&(p.prestato),p.
data,)!=EOF)
        {
            temp=cons(p,temp);
            letti++;
        }
        (*l)=temp;
        fclose(fp);
        return 1;
    }
}

```

```

}

int conta(list l)
{
    int i=0;
    list t=l;
    libro temp;
    char *data="2007/04/01";
    while(!isEmpty(t))
    {
        temp=head(t);
        if((strcmp(temp.data,data))<0 && temp.prestato!=0)
        {
            ++i;
        }
        t=tail(t);
    }
    return i;
}

/* opzionale */
void showList(list l)
{
    list t=l;
    libro temp;

    while(!isEmpty(t))
    {
        temp=t->value;
        printf("ID: %s, coordinate: %s, data: %s\n",
temp.id,temp.coordinate,temp.data);
        t=t->next;
    }
}

```

## main.c

```

#include <stdio.h>
#include "funzioni.h"

int main()
{
    int i=0;
    libri p;
    list l=emptyList();
    if((leggi("LIBRI.TXT",&l))==0)
    {
        printf("Errore nell'apertura del file");
        exit(1);
    }

    /* da qui */
    printf("\nInizio Elenco\n");
    showList(l);
    printf("Fine Elenco\n");
    /* a qui opzionale */

    i=conta(l);
    printf("\nLibri prestati prima del 01/04/2007: %d \n",i);
}

```

```
    return 0;  
}
```

## LIBRI.TXT

```
7FYHR8R0 dfy 0 2007/03/27  
84H32NDI gey 0 2006/06/28  
HFUH334D nhx 1 2006/04/30  
JIUROEJD fgr 0 2006/06/26  
478EEH33 caz 1 2006/07/15
```

## 2. SQL (9 punti)

```
select libro.titolo  
from libro, corso, consigliato
```

```
where corso.anno=2 and corso.codice=consigliato.codCorso and  
consigliato.ISBN=libro.ISBN
```

## SCHEMA LOGICO: CASA EDITRICE (9 punti)

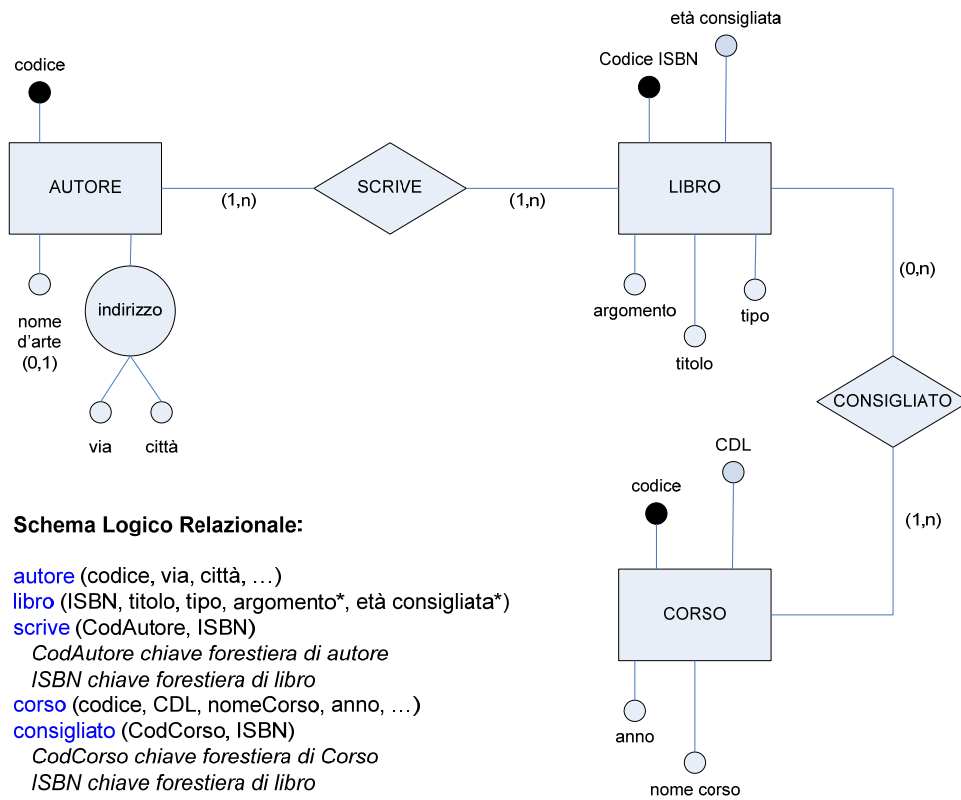
### Entità e Relazioni:

Entità: libro, testo, corso, autore

Relazioni: scrive, consigliato

**Errore grave! inserire l'entità "casa editrice", poiché contiene una sola istanza**

esiste una sola generalizzazione sulle entità libro, libroDiTesto, libroGenerico che viene tradotta nel modello logico mediante collasso verso l'alto.



**Schema Logico Relazionale:**

**autore** (codice, via, città, ...)

**libro** (ISBN, titolo, tipo, argomento\*, età consigliata\*)

**scrive** (CodAutore, ISBN)

*CodAutore chiave forestiera di autore*

*ISBN chiave forestiera di libro*

**corso** (codice, CDL, nomeCorso, anno, ...)

**consigliato** (CodCorso, ISBN)

*CodCorso chiave forestiera di Corso*

*ISBN chiave forestiera di libro*