

Fondamenti di Informatica A

Ing. Elettronica e delle Telecomunicazioni

Esercitazione 4 - Soluzioni

7-8 Novembre 2006

Record

ESERCIZIO n° 1:

Scrivere un programma che acquisisca dallo standard input i dati relativi all'archivio di una biblioteca. In particolare, ogni elemento dell'archivio rappresenta un libro ed e' caratterizzato dalle seguenti informazioni:

- **codice** (un intero che individua univocamente il libro)
- **titolo**
- **autore**
- **casa_editrice**
- **stato**: un valore intero (**tag**) che indica se il libro e' in prestito (valore 0) o disponibile (valore 1).

Una volta inizializzato l'archivio con valori forniti da standard input, il programma deve:

1. Dati in ingresso titolo e autore di un libro presente in archivio, stampare tutte le informazioni relative al libro dato.
2. Dato il nome A di un autore, il programma dovra' contare e stampare i dati di tutti i libri disponibili al prestito scritti da A. Dovra' inoltre stampare il numero totale di essi.
3. Stampare il contenuto dell'intero archivio.

```
#include <stdio.h>
#include <string.h>

#define N 80

typedef struct Libro { int codice;
                      char titolo[N];
                      char autore[N];
                      char casa_editrice[N];
                      int stato;
                    } Libro;

void main()
{int n,i;
 Libro archivio[N];
 printf("Quanti libri vuoi inserire ? ");
 scanf("%d",&n);

 for (i=0; i<n; i++)
 {printf("Libro #.%d : codice ",i);
  int bool;
  do
  {scanf("%d",&archivio[i].codice);
   bool=0;
   for (int j=0; j<i; j++)
    if (archivio[i].codice== archivio[j].codice)
    {bool=1;
     printf("Codice esistente\nLibro #.%d : codice ",i);
     break;
    }
  }
  while (bool==1);
```

```
fflush(stdin);
 printf("Libro #.%d : titolo ",i);
 gets(archivio[i].titolo);
 printf("Libro #.%d : autore ",i);
 gets(archivio[i].autore);
 printf("Libro #.%d : casa editrice ",i);
 gets(archivio[i].casa_editrice);
 printf("Libro #.%d : stato ",i);
 scanf("%d",&archivio[i].stato);
 }

 int trovato, cont;
 char c;
 char titolo[N];
 char autore[N];
 while (c!='0')
 {printf("\nScegliere un'opzione : \n 1) Scelta libro\n 2)
   Scelta autore\n 3) Stampa archivio\n 0) EXIT\n");
  fflush(stdin);
  scanf("%c",&c);

  switch (c)
  {case '1' :
   fflush(stdin);
   printf("Titolo del libro : ");
   gets(titolo);
   printf("Autore del libro : ");
   gets(autore);
   trovato=0;
   for (i=0; (i<n) && (!trovato); i++)
    if (!(strcmp(archivio[i].titolo,titolo)) && (!strcmp(archivio[i].autore,autore)))
    {trovato=1;
     printf("Codice : %d\n",archivio[i].codice);
     printf("Titolo : %s\n",archivio[i].titolo);
     printf("Autore : %s\n",archivio[i].autore);
     printf("Casa : %s\n",archivio[i].casa_editrice);
     printf("Stato : %d\n",archivio[i].stato);
    }
   if (!trovato) printf("Libro non in archivio\n");
   break;

  case '2' :
   fflush(stdin);
   printf("Autore del libro : ");
   gets(autore);
   cont=0;
   for (i=0; i<n; i++)
    if (!(strcmp(archivio[i].autore,autore)) && (archivio[i].stato))
    {cont++;
     printf("%s\n",archivio[i].titolo);
    }
   printf("Sono disponibili al prestito %d libri
   scritti da %s\n", cont, autore);
   break;

  case '3' :
   for (i=0; i<n; i++)
    {printf("%d\n",archivio[i].codice);
     printf("%s\n",archivio[i].titolo);
     printf("%s\n",archivio[i].autore);
     printf("%s\n",archivio[i].casa_editrice);
     printf("%d\n",archivio[i].stato);
    }
   break;
 }
 }
```

ESERCIZIO n° 2:

Realizzare un programma che legga da standard input l'elenco dei risultati relativi ad una prova scritta di esame. In particolare, per ogni studente che ha partecipato alla prova scritta, vengono date:

- **Matricola:** un intero che rappresenta il numero di matricola dello studente ed individua quindi univocamente lo studente;
- **Cognome:** una stringa contenente il cognome dello studente;
- **Nome:** una stringa contenente il nome dello studente;
- **Voto:** un intero che rappresenta il voto conseguito dallo studente nello scritto. In particolare, si assuma che il voto appartenga all'intervallo [0,30].

Si supponga che gli studenti partecipanti allo scritto non possano essere più di 100.

Una volta letti i dati e collocati in un'opportuna struttura **E**, il programma deve:

1. Stampare cognome, nome e voto di tutti gli studenti che hanno superato la prova (quelli, cioè, che hanno ottenuto un voto maggiore o uguale a 18).
2. Stampare la media aritmetica di tutti i voti riportati dagli studenti promossi.
3. Stampare la percentuale degli studenti bocciati (quelli, cioè, che hanno ottenuto un voto minore di 18).

```
#include <stdio.h>
#include <string.h>

#define N 100

typedef struct Studente { int matricola;
                        char cognome[N];
                        char nome[N];
                        int voto;
                        } Studente;

void main()
{int n,i;
 Studente E[N];
 printf("Quanti studenti vuoi inserire ? ");
 scanf("%d",&n);

 for (i=0; i<n; i++)
 {printf("Studente #.%d : matricola ",i);
  int bool;
  do
  {scanf("%d",&E[i].matricola);
   bool=0;
   for (int j=0; j<i; j++)
    if (E[i].matricola== E[j].matricola)
    {bool=1;
     printf("Matricola esistente\nStudente #.%d : matricola ",i);
     break;
    }
  }
  while (bool==1);

  fflush(stdin);
  printf("Studente #.%d : cognome ",i);
  gets(E[i].cognome);
  printf("Studente #.%d : nome ",i);
  gets(E[i].nome);
  printf("Studente #.%d : voto (0-30) ",i);
  do
  {scanf("%d",&E[i].voto);
   while ((E[i].voto<0) || (E[i].voto>30));
  }
 }
```

```
int cont=0, media=0;
for (i=0; i<n; i++)
  if (E[i].voto>=18)
  {printf("Matricola : %d\n",E[i].matricola);
   printf("Cognome : %s\n",E[i].cognome);
   printf("Nome : %s\n",E[i].nome);
   printf("Voto : %d\n\n",E[i].voto);
   cont++;
   media+=E[i].voto;
  }

printf("La media dei promossi e' %f\n", (float) media/cont);
printf("La percentuale di bocciati e' %f\n",(n-cont) / ((float)n)*100);
}
```