

**Politecnico di Milano - Anno Accademico 2002-2003 - Informatica C - prof. R.Martucci**  
**Seconda Prova in Itinere e Lab- 10 Luglio 2003 – Elaborato B**

NOME COGNOME	
Matricola	

Tot/Punti	1	2	3	3 Bonus
/ <b>18</b>	/ 8	/ 7	/ 3	/ 2

**Es.1** - Definire una struttura per descrivere il percorso di una gara ciclistica a tappe (città partenza, città di arrivo, lunghezza tappa, variazione altimetrica). Scrivere poi un programma che dichiari un array di 10 strutture di quel tipo, (1p) poi legga i dati di 10 tappe caricando le relative strutture nell'array (1p) Al termine della fase di immissione dati, il programma visualizza per ogni tappa: il nome della città di arrivo (1p), il progressivo della distanza percorsa dall'inizio della gara (2p) e, alla fine, quale è l'altitudine massima tra le varie tappe (3p).

```
#include <stdio.h>
#define N 10
typedef struct {
    char part[20], arr[20];
    double lung;
    double alt;
} TAPPA;

void main()
{
    TAPPA gara[N];
    int i;
    double dist=0, altmax=0;

    for(i=0; i<N; i++)
    {
        printf("Inserisci partenza: ");
        scanf("%s", gara[i].part);
        printf("Inserisci arrivo: ");
        scanf("%s", gara[i].arr);

        printf("Inserisci lung: ");
        scanf("%lf", &(gara[i].lung));
        printf("Inserisci diff altim: ");
        scanf("%lf", &(gara[i].alt));
    }

    for(i=0; i<N; i++)
    {
        printf("Tappa %d ", i+1);
        printf("arrivo %s ", gara[i].arr);
        dist += gara[i].lung;
        printf("progr. Dist.: %f\n", dist);
        if (gara[i].alt > altmax)
            altmax = gara[i].alt;
    };
    printf("max alt= %lf", altmax);
}
```

**Es.2** - Due file di testo inp1.txt ed inp2.txt contengono un numero imprecisato ma eguale di righe di testo lunghe al massimo 80 caratteri ciascuna. Ogni riga contiene una stringa priva di spazi. Scrivere un programma che tenta di aprire in lettura i due file segnalando un eventuale insuccesso (1p), poi, finché i file non terminano, li legge in parallelo riga per riga e conta quante righe sono eguali (2p). Il programma, quando incontra in un file una riga con meno di 6 caratteri, stampa un messaggio contenente le righe (2p) e aggiorna un contatore di righe corte (1p). Quindi stampa il numero di righe eguali che ha incontrato e il numero di righe corte; alla fine chiude i file aperti. (1p)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define LIM 6
#define LR 80

void main()
{
    char r1[LR+1], r2[LR+1];
    FILE *i1, *i2;
    int conta=0, corte=0;

    if((i1=fopen("inp1.txt", "r"))==NULL)
        {printf("impossibile aprire inp1.txt\n"); exit(-1); }
    if((i2=fopen("inp2.txt", "r"))==NULL)
        {printf("impossibile aprire inp2.txt\n"); fclose(i1); exit(-1); }
```

**Politecnico di Milano - Anno Accademico 2002-2003 - Informatica C - prof. R.Martucci**  
**Seconda Prova in Itinere – 10 Luglio 2003 – Elaborato B**

NOME COGNOME	
Matricola	

```

while(fscanf(i1,"%s",r1)!=EOF)
    {fscanf(i2,"%s",r2);
      if (strlen(r1) < LIM || strlen(r2) < LIM)
        {corte++;
          printf("riga corta %s-%s\n",r1,r2); }

      if (strcmp(r1,r2) == 0)
        conta++;
    }

printf("ho trovato %d righe eguali\n",conta);
printf("riga con meno di %d caratteri %d\n",LIM,corte);

fclose(i1); fclose(i2);
}

```

**Es.3-** Scrivere una funzione **funz1** che riceve come parametro un array di N interi e calcola (ritornandone il valore) la somma dei quadrati dei valori ivi contenuti (**1p**). Scrivere poi un main che carica un array di N interi leggendo i valori da tastiera (**1p**) e successivamente chiama sull'array la funzione **funz1** sopra descritta e visualizza il risultato da essa ritornato (**1p**). I punti sono aumentati (**bonus extra**) se l'array è manipolato completamente con la notazione a puntatori.

```

#include <stdio.h>
#define N 3

float funz1(int * a )
{
    int i;
    float tot=0;

    for(i=0; i<N; i++)
        tot+= (*(a+i)) * (*(a+i));
    return tot;
}

void main()
{
    int i;
    int a[N];
    for(i=0; i<N; i++)
    {
        printf("Inserisci valore n.%d ",i);
        scanf("%d",a+i);
    }
    printf("La media e': %f\n",funz1(a));
}

```

```

#include <stdio.h>
#define N 3

float funz1(int a[])
{
    int i;
    float tot=0;
    for(i=0; i<N; i++)
        tot+= (a[i]*a[i]);
    return tot;
}

void main()
{
    int i;
    int a[N];
    for(i=0; i<N; i++)
    {
        printf("Inserisci valore n.%d ",i);
        scanf("%d",a+i);
    }
    printf("La media e': %f\n",funz1(a));
}

```

**Politecnico di Milano - Anno Accademico 2002-2003 - Informatica C - prof. R.Martucci**  
**Seconda Prova in Itinere – 10 Luglio 2003 – Elaborato B**

NOME COGNOME	
Matricola	

Tot/Punti	LAB 1	LAB 2
/5	/3	/2

**Lab.1** - Dato il seguente frammento di programma C individuare gli 8 errori sintattici (quelli rilevabili dal compilatore).

```
#include <stdio.h>
#define DIM 5;
typedef struct {int a,b} EL;
int funz (EL v)
{ int i;
  for(i=0; i<DIM-1; i++)
    v(i) = v[i+1];
  v[1].a = v[0].b;
  return(v[0].a+v[0].b);
}
void main()
{ FILE *f;
  EL el[DIM];
  int i, v;
  if((f==fopen("m.txt", "w"))==NULL)
    exit(-1);
  for (i=0; i<DIM; i++)
    {el[i].a = i
     el[i].b = i+10;}
  v = funz(&el);
  for(i=0; i<DIM; i++)

fprintf("%d\t%d\n", el[i].a, el[i].b);
  fclose(f);
}
#include <stdio.h>
```

```
#include <stdlib.h>
#define DIM 5 /*non va il ;*/
typedef struct {int a,b i } EL;
int funz (EL v[])
{ int i;
  for(i=0; i<DIM-1; i++)
    v[i] = v[i+1];
  v[1].a = v[0].b;
  return(v[0].a+v[0].b);
}
void main()
{ FILE *f;
  EL el[DIM];
  int i, v;
  if((f=fopen("m.txt", "w"))==NULL)
    exit(-1);
  for (i=0; i<DIM; i++)
    {el[i].a = i i
     el[i].b = i+10;}
  v = funz(el);
  for(i=0; i<DIM; i++)
  fprintf(f, "%d\t%d\n", el[i].a, el[i].b);
  fclose(f);
}
```

**Lab. 2** - Il seguente programma vuole determinare il numero intero I più prossimo ad un numero reale R dato. Il programma contiene uno o più errori logici che ne impediscono il funzionamento. Correggere il programma in modo che calcoli quanto richiesto (provare con i seguenti valori: R=3.67 R=3.0 R=-3.76 R=-1.0 R=-0.6 R=0.0)

```
#include <stdio.h>
void main()
{int I;
 double R;
 scanf("%lf", &R);
 if(R > 0)
  for(I=0; I<=R; I++);
 else
  if (R < 0)
   { for (I=0; I<=-R; I++);
     I = -I; }
  printf("%d\n", I);
}
```

```
#include <stdio.h> /*doppia versione */
void main()
{int I=0;
 double R;
 scanf("%lf", &R);
 R = R + .5;
 if(R > 0)
  for(I=0; I<R; I++) /*senza ==*/;
 else
  if (R < 0)
   { for (I=0; I<=-R; I++);
     I = -(I-1); }
 I = I - 1;
 printf("%d\n", I);} /*continua pag 4*/
```

**Politecnico di Milano - Anno Accademico 2002-2003 - Informatica C - prof. R.Martucci**  
**Seconda Prova in Itinere – 10 Luglio 2003 – Elaborato B**

NOME COGNOME	
Matricola	

```
/*considerando l'approssimazione
all'intero più grande*/
#include <stdio.h>
void main()
{int I=0;
 double R;
 scanf("%lf",&R);
 if(R > 0)
   for(I=0; I<R; I++) /*senza =*/;
 else
   if (R < 0)
     { for (I=0; I<=-R; I++);
       I = -(I-1);}
 printf("%d\n",I);
}
```