

Fondamenti di Informatica L-A (A.A. 2002/2003) - Ingegneria Informatica
Prof.ssa Mello & Prof. Bellavista – Prova d'Esame di Lunedì 08/01/2003 – durata 2h:30m
COMPITO A

ESERCIZIO 1 (10 punti)

È dato un file di testo **meteo.txt** che contiene i dati relativi alle condizioni atmosferiche registrate nelle città campione durante il giorno 25/12/2002. Più precisamente, ogni riga contiene, nell'ordine:

- nome città (non più di 20 caratteri senza spazi), uno e un solo spazio di separazione;
- temperatura minima in gradi centigradi (numero intero), uno e un solo spazio di separazione;
- temperatura massima in gradi centigradi (numero intero), uno e un solo spazio di separazione;
- descrizione (non più di 12 caratteri senza spazi), terminatore di riga.

Ad esempio, **meteo.txt**:

ALGHERO 5 16 VARIABILE
ANCONA 7 9 NUVOLOSO
AOSTA 0 4 NEVE
BARI 10 14 VARIABILE
BOLOGNA 7 9 PIOGGIA

.....

Si scriva una procedura che riceva come parametro di ingresso un file di testo **F** di strutture **prelievo** contenenti i dati meteo e restituisca come parametro di uscita un vettore **Vett** contenente le strutture di **F** relative ai prelievi effettuati nelle città con temperature (minime e massime) comprese tra i valori **MIN** e **MAX**, ed il loro numero **N**. (4 punti)

```
void find(FILE * F, prelievo Vett[], int MIN, int MAX, int *N);
```

Si scriva inoltre un programma C che inserisca in un vettore **V1** (supposto di dimensione massima **DIM 100**) tutti i prelievi contenuti nel file **meteo.txt** effettuati in città con temperature comprese tra i 7 e i 10 gradi centigradi utilizzando la procedura **find()** precedentemente definita. Il programma deve inoltre stampare a terminale gli elementi di **V1** relativi alle città senza precipitazioni (né neve né pioggia). (6 punti)

ESERCIZIO 2 (5 punti)

Si scriva una funzione ricorsiva **g()** che data in ingresso una lista di interi ed un numero **num**, restituisca in uscita una nuova lista contenente esclusivamente gli elementi maggiori di **num**, decrementati di un valore uguale a **num**. Ad esempio, se invocata con **l=[11,0,7,8,3,-2,15]** e **num=7**, la funzione **g()** deve restituire la lista **[4,1,8]**. Si supponga di possedere il tipo di dato astratto **list** e le sue operazioni **PRIMITIVE** definite durante il corso (che quindi possono NON essere riportate nella soluzione).

ESERCIZIO 3 (7 punti)

Dato il programma sorgente C seguente, si indichino i valori stampati a tempo di esecuzione, motivando la risposta data. Indicare inoltre quali sono i blocchi in cui è visibile la variabile **M**, sempre motivando la risposta.

```
#include <stdio.h>
#define N 7
#define FALSE 0
#define TRUE 1

typedef int vettore[N];
void b(vettore,int,int);
```

```

main ()
{
    int i; vettore a;
    for (i = 0; i < N; i++){
        a[i]=i/2.0f;
        if (a[i]-a[0]) a[i]=i;
    }
    b(a, N/2, N-1);
    for (i = N-1; i >= 0; --i) printf ("%d\n", a[i]);
}

```

```

void b(vettore v, int iniz, int fine)
{
    int I; float temp, M=0.0f;
    for (I = iniz; I < fine; I++){
        if (v[I] < v[I+1]){
            temp = v[I]-1;
            v[I] = v[I+1];
            v[I+1] = temp+1;
        }
    }
}

```

Esercizio 4 (6 punti)

Si consideri la seguente funzione F:

```

int G(double x){
    if (x>0) {x-=2; return G(x)/2;}
    else return 1;
}

```

Si scriva il risultato della funzione quando invocata come **G(6)** e si disegnino i corrispondenti record di attivazione.

Esercizio 5 (2 punti)

Si consideri la grammatica G con scopo S e simboli terminali { **ciao**, “”, (,), **printf** }

S::= A

A::=) | printf C | ” A

B::= ciao C | “ B

C::= A | (B

Si dica se la stringa **printf(“ciao”)** è sintatticamente corretta rispetto a tale grammatica e se ne mostri la derivazione left most.

Esercizio 6 (2 punti)

Un elaboratore rappresenta i numeri interi su 8 bit dei quali 7 sono dedicati alla rappresentazione del modulo del numero e uno al suo segno. Indicare come viene svolta la seguente operazione aritmetica e determinarne il risultato, traslandolo poi in decimale per la verifica:

111-92

Fondamenti di Informatica L-A (A.A. 2002/2003) - Ingegneria Informatica
Prof.ssa Mello & Prof. Bellavista – Prova d'Esame di Lunedì 08/01/2003 – durata 2h:30m
COMPITO B

ESERCIZIO 1 (10 punti)

È dato un file di testo **canzoni.txt** che contiene i dati relativi alla playlist musicale relativa alla colonna sonora di un film. Più precisamente, ogni riga contiene, nell'ordine:

- nome cantante (non più di 12 caratteri senza spazi), uno e un solo spazio di separazione;
- titolo (non più di 15 caratteri senza spazi), uno e un solo spazio di separazione;
- durata in secondi (numero intero positivo), uno e un solo spazio di separazione;
- genere (non più di 12 caratteri senza spazi), terminatore di riga.

Ad esempio, **canzoni.txt**:

```
PORTISHEAD COWBOYS 278 DRUM&BASE
NIRVANA LITHIUM 256 ROCK
GERSHWIN SUMMERTIME 290 JAZZ
EMINEM KIM 377 RAP
RADIOHEAD CREEP 267 ROCK
```

.....

Si scriva una procedura che riceva come parametro di ingresso un file di testo **F** di strutture **brano** contenenti i dati sulle canzoni e restituisca come parametro di uscita un vettore **Vett** contenente le strutture di **F** relative ai brani la cui durata sia compresa nell'intervallo definito da **MIN** e **MAX**, ed il loro numero **N**. (4 punti)

```
void play(FILE * F, brano Vett[], int MIN, int MAX, int *N);
```

Si scriva inoltre un programma C che inserisca in un vettore **V1** (supposto di dimensione massima **DIM 100**) tutti i brani contenuti nel file **canzoni.txt** di durata compresa tra 260 e 300 secondi utilizzando la procedura **play()** definita al punto precedente. Il programma deve inoltre stampare a terminale tutti gli elementi di **V1** relativi al genere **ROCK** (6 punti).

ESERCIZIO 2 (5 punti)

Si scriva una funzione ricorsiva **f(list l, int num)** che data in ingresso una lista di interi e un numero **num**, restituisca in uscita una nuova lista non contenente gli elementi di valore uguale a **num** e avente gli altri elementi con i valori cambiati di segno. Ad esempio, se invocata con **l=[0,7,3,-2,7,1]** e **num=7**, la funzione **f()** deve restituire **[0,-3,2,-1]**.

Si supponga di possedere il tipo di dato astratto **list** e le sue operazioni **PRIMITIVE** definite durante il corso (che quindi possono NON essere riportate nella soluzione).

ESERCIZIO 3 (7 punti)

Dato il programma sorgente C seguente, si indichino i valori stampati a tempo di esecuzione, motivando la risposta data. Indicare inoltre quali sono i blocchi in cui è visibile la variabile **M**, sempre motivando la risposta.

```
#include <stdio.h>
#define N 7
#define FALSE 0
#define TRUE 1
typedef int vettore[N];

void b(int,int,vettore);
```

```

main ()
{
int i;
    vettore a;
    for (i = 0; i < N; i++){
        a[i]=i*2.0f;
        if (!(a[i]-a[0])) a[i]=i;
    }
    b(0, N-1, a);
    for (i = N-1; i >= 0; i--=1) printf ("%d , ", a[i]);
}

```

```

void b(int iniz, int fine,vettore v)
{
int I;
    float temp,M;
    for (I = iniz; I < fine; I++){
        if (v[I] < v[I+1]){
            temp = v[I]--;
            v[I] = v[I+1];
            v[I+1] = temp++;
        }
    }
}

```

Esercizio 4 (6 punti)

Si consideri la seguente funzione F:

```

int F(float y){
    if (y>0) {y-=3; return F(y)/2;}
    else return 8;
}

```

Si scriva il risultato della funzione quando invocata come F(9.0) e si disegnino i corrispondenti record di attivazione.

Esercizio 5 (2 punti)

Si consideri la grammatica G con scopo S e simboli terminali { **pie**de, **den**te, **in**, **oc**chio, **tes**ta, **per** }

S::= A

A::= **pie**de | **oc**chio C | **den**te

B::= A C | **den**te

C::= A | **per** B

Si dica se la stringa **oc**chio **per** **oc**chio **den**te **per** **den**te è sintatticamente corretta rispetto a tale grammatica e se ne mostri la derivazione left most.

Esercizio 6 (2 punti)

Un elaboratore rappresenta i numeri interi su 8 bit dei quali 7 sono dedicati alla rappresentazione del modulo del numero e uno al suo segno. Indicare come viene svolta la seguente operazione aritmetica e determinarne il risultato, traslandolo poi in decimale per la verifica:

$$19+100$$

SOLUZIONE COMPITO A

ESERCIZIO 1

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define DIM 100

typedef struct {char citta[21]; int min; int max; char descr[13];}
    prelievo;

void find(FILE * F, prelievo Vett[], int MIN, int MAX, int *N) {
    prelievo in; *N=0;

    while (fscanf(F,"%s %d %d %s\n", in.citta, &in.min, &in.max, in.descr)
        != EOF) {
        if ( (in.min>=MIN)  && (in.max<=MAX) )
            {
                Vett[*N]=in;
                *N= *N+1; }
    }
}

main() {
    prelievo V1[DIM];
    int Num, i=0;
    FILE* f;

    if ((f=fopen("c:\\meteo.txt", "r"))==NULL) {
        printf("Il file non esiste!"); exit(1); }
    find(f, V1, 7, 10, &Num);
    for (i=0; i<Num; i++)
        if( (strcmp(V1[i].descr,"PIOGGIA")!=0) &&
            (strcmp(V1[i].descr,"NEVE")!=0) )
            printf("%s %d %d %s\n", V1[i].citta, V1[i].min, V1[i].max,
                V1[i].descr);
    fclose(f);
}

/* La funzione int strcmp(char* s1,char* s2) restituisce 0 se e solo se le
    stringhe s1 e s2 sono identiche */
```

ESERCIZIO 2

```
list g(list l, int num) {
    if empty(l) return emptylist();
    else if ( (head(l)>num) )
        return cons(head(l)-num, g(tail(l), num));
    else return g(tail(l), num);
}
```

ESERCIZIO 3

Il programma stampa:

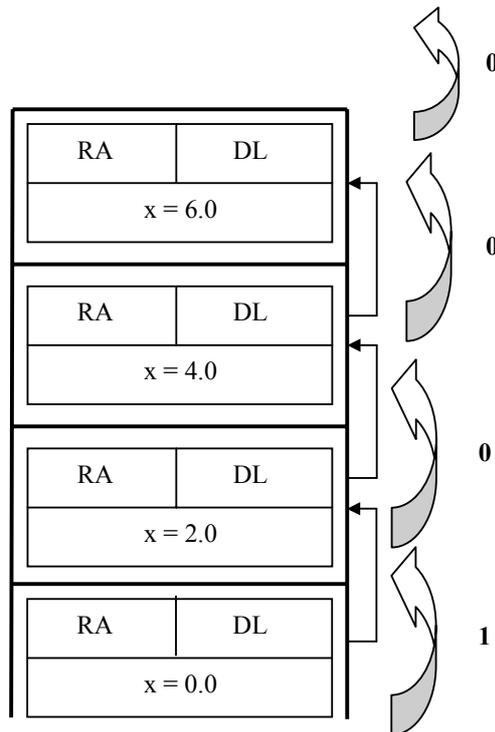
3 6 5 4 2 0 0 (con tutti i numeri su righe differenti)

Infatti il vettore a viene inizializzato con il seguente contenuto: {0,0,2,3,4,5,6}. Successivamente viene invocata la procedura b alla quale si passa per riferimento il vettore a. In particolare, la procedura b attraverso un ciclo (for) scambia la posizione del valore 3 nel vettore. Alla fine del ciclo il valore 3 sarà posizionato come ultimo elemento, e gli altri elementi scaleranno ordinatamente di una posizione. Infine il vettore viene stampato al contrario.

La variabile M è una variabile locale alla funzione b, perciò il suo campo di visibilità è limitato alla sola funzione b.

ESERCIZIO 4

La funzione restituisce il valore 0.



Esercizio 5

La stringa è sintatticamente corretta. Derivazione *left most*:

```
S -> A -> printf C -> printf( B -> printf(" B -> printf("ciao C ->
printf("ciao A -> printf("ciao" A -> printf("ciao")
```

Esercizio 6

```
111-> 0  1101111
-92-> 1  1011100
```

Tra i moduli dei numeri si esegue una sottrazione:

```
  1101111
-  1011100
-----
  0010011
```

che vale 19 in base dieci.

SOLUZIONE COMPITO B

ESERCIZIO 1

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define DIM 100

typedef struct {char cantante[13]; char titolo[16]; int durata; char
    genere[13];} brano;

void play(FILE * F, brano Vett[], int MIN, int MAX, int *N) {
    brano in; *N=0;

    while (fscanf(F,"%s %s %d %s\n", in.cantante, in.titolo, &in.durata,
in.genere) != EOF) {
        if ( (in.durata>=MIN) && (in.durata<=MAX) )
            {
                Vett[*N]=in;
                *N= *N+1; }
    }
}

main() {
    brano V1[DIM];
    int Num, i=0;
    FILE* f;

    if ((f=fopen("c:\\canzoni.txt", "r"))==NULL) {
        printf("Il file non esiste!"); exit(1); }
    play(f, V1, 260, 300, &Num);
    for (i=0; i<Num; i++)
        if( (strcmp(V1[i].genere,"ROCK")==0) )
            printf("%s %s %d %s\n", V1[i].cantante, V1[i].titolo, V1[i].durata,
V1[i].genere);
    fclose(f);
}

/* La funzione int strcmp(char* s1,char* s2) restituisce 0 se e solo se le
stringhe s1 e s2 sono identiche */
```

ESERCIZIO 2

```
list f(list l, int num) {
    if empty(l) return emptylist();
    else if(head(l)==num) return f(tail(l), num);
    else return cons(-head(l), f(tail(l), num));
}
```

ESERCIZIO 3

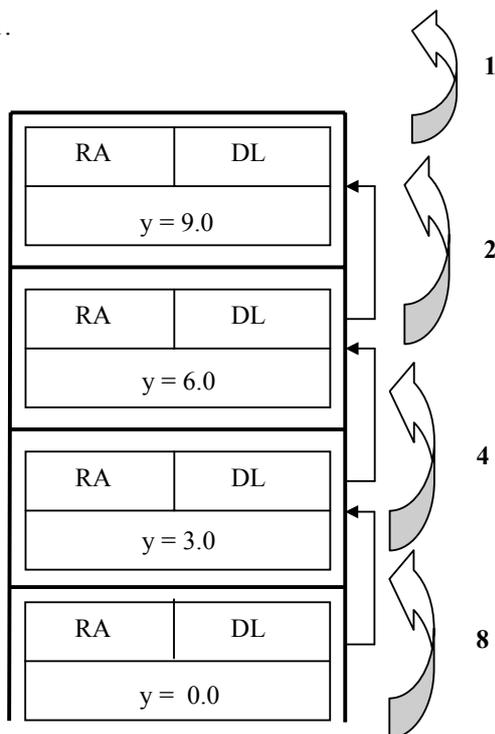
Il programma stampa:
0, 12, 10, 8, 6, 4, 2,

Infatti il vettore v viene inizializzato con il seguente contenuto: {0,2,4,6,8,10,12}. Successivamente viene

invocata la procedura b alla quale si passa come parametro il vettore a. In particolare, la procedura b attraverso un ciclo (for) scambia la posizione del primo elemento del vettore (0) con i successivi elementi. Alla fine del ciclo, lo 0 sarà posizionato nell'ultima posizione del vettore. Poi il vettore viene stampato al contrario. La variabile M è una variabile locale alla funzione b, perciò il suo campo di visibilità è limitato alla sola funzione.

ESERCIZIO 4

La funzione restituisce il valore 1.



Esercizio 5

La stringa è sintatticamente corretta.

Derivazione left most:

S -> A -> occhio C -> occhio per B -> occhio per AC -> occhio per occhio C
 C -> occhio per occhio A C -> -> occhio per occhio dente C -> occhio per occhio dente per B -> occhio per occhio dente per dente

Esercizio 6

19-> 0 0010011
 100-> 0 1100100

Tra i moduli dei numeri si esegue una somma:

```

    0010011
  + 1100100
  -----
    1110111
  
```

che vale 119 in base dieci.