

Fondamenti di Informatica - Ing. Civile/Edile - Dott. Penzo
Soluzione compito A - 19/10/2001

Esercizio 1

- 1) Con il metodo delle divisioni successive, al numero 127 in base 10 corrisponde il numero 1111111 in base 2. Per la conversione nelle basi 8 e 16 si possono applicare due metodi: nuovamente il metodo delle divisioni successive a partire da 127 oppure il raggruppamento in blocchi di 3 e di 4 cifre, rispettivamente per la base 8 e la base 16, del numero binario ottenuto in precedenza. I numeri che si ottengono sono: 177 in base 8 e 7F in base 16.
- 2) Occorre eseguire il procedimento inverso del complemento a 2 di un numero binario: sottrarre 1 al numero iniziale e complementare a 1 tutte le cifre. Si ottiene così il modulo del numero negativo iniziale. La conversione in base 10 di tale modulo consente di determinarne il valore decimale. Dal numero 10010110 sottraendo 1 si ottiene 10010101 che, complementato a 1, corrisponde a 01101010, ossia 106. Il numero iniziale equivale quindi a -106 in base 10.

Esercizio 2

Le istruzioni contenute all'interno del ciclo while del main vengono eseguite 3 volte. Infatti: per $i=0$ si ha $V[0]=1$ e $0<1$ (la condizione del while è soddisfatta); per $i=1$ si ha $V[1]=2$ e $1<2$; per $i=2$ si ha $V[2]=3$ e $2<3$; infine per $i=3$ si ha $V[3]=2$ e $3<2$ non è soddisfatta e si esce dal ciclo.

Per ogni iterazione viene chiamata la funzione F sul vettore A (che è l'unico che viene modificato!) con secondo parametro pari a 1, 2 e 3, rispettivamente nella prima, nella seconda e nella terza iterazione (per quanto detto sopra).

La prima esecuzione di F ha quindi $j=1$ e il for interno viene eseguito 2 volte: per $i=0$ e per $i=1$, dopodiché si esce perché $2 \leq 1$ è falso. Per $i=0$ $A[0]$ (il parametro formale è V ma quello attuale è A, quindi si agisce su A) assume il valore $A[0] + (0+1) = 1$.

Per $i=1$ ad $A[1]$ viene assegnato $A[1] + (1+1) = 2$. All'uscita del ciclo for i vale 2 e prima di effettuare l'istruzione di return viene decrementato il valore di i ($--i$) per cui viene restituito $A[1]$ ossia 2. La prima printf del main stampa quindi il valore 2.

Le rimanenti iterazioni del while del main si comportano similmente, con le chiamate alla funzione F con $j=2$ e con $j=3$, modificando ulteriormente il vettore A. Nel complesso il programma stampa:

2	4	6	(valori della funzione F)	
6	9	9	6	0 (vettore A modificato)
1	2	3	2	1 (vettore V inalterato)

Esercizio 3

```
#include<stdio.h>
#define N 30

typedef struct {
    char nome[20];
    char allenatore[30];
    int punteggio;
} squadra;

void CaricaVettore(squadra C[], int *n);
void AggiornaCampionato(char filename[], int num, squadra C[]);
void AggiornaVettore(squadra C[], int num, char *nome, int punti);
void Campione(squadra C[], int num, char *champion);

main()
{
    int numelem = 0;
    squadra CAMPIONATO[N];
    char *champion;

    CaricaVettore(CAMPIONATO,&numelem);
    AggiornaCampionato("PARTITE.TXT",numelem,CAMPIONATO);
    Campione(CAMPIONATO,numelem,champion);
}

void CaricaVettore(squadra C[], int *n)
{
    FILE *fp;

    fp = fopen("SQUADRE.DAT","rb");
    if (fp == NULL) printf("Errore di apertura file!\n");
    else while(!feof(fp)){
        fread(&C[*n],sizeof(squadra),1,fp);
        (*n)++;
    }
    fclose(fp);
}

void AggiornaCampionato(char filename[], int num, squadra C[])
{
    FILE *fp;
    char *squadra_casa, *squadra_ospite;
    int goal_casa, goal_ospite;

    fp = fopen(filename, "r");
    if (fp == NULL) printf("Errore di apertura file!\n");
    else while(!feof(fp)){
        fscanf(fp,"%s%s%d%d",squadra_casa,squadra_ospite,&goal_casa,&goal_ospite);
        if (goal_casa > goal_ospite)
            AggiornaVettore(C,num,squadra_casa,3);
        else if (goal_casa < goal_ospite)
            AggiornaVettore(C,num,squadra_ospite,3);
    }
}
```

```

        else {
            AggiornaVettore(C,num,squadra_casa,1);
            AggiornaVettore(C,num,squadra_ospite,1);
        }
    }
}

void AggiornaVettore(squadra C[], int num, char *nome, int punti)
{
    int i;
    int trovato = 0;

    while((i < num) && !trovato){
        if (!strcmp(C[i].nome, nome)){
            trovato = 1;
            C[i].punteggio += punti;
        }
        i++;
    }
}

void Campione(squadra C[], int num, char *champion)
{
    Bubblesort(C,num);
    strcpy(champion,C[0].nome);
}

```