

# Fondamenti di Informatica - Ing. Civile/Edile - Dott. Penzo

## Soluzione compito 19/07/2002

### Esercizio 1

Per i passaggi intermedi vedere le dispense.

22 in base 2: 00010110  
0.75 in base 2: .11000....

22.75 in base 2: 10110.11000....  
normalizzando: .10110110 mantissa e 00000101 esponente  
rappresentaz. : .00110110 mantissa e 00000101 esponente

13 in base 2: 00001101  
9 in base 2: 00001001

13 + 9 = 00010110 = 22 in base 10  
22 in floating point normalizzato: .10110000 mantissa e 00000101 esponente  
22 in rappresentazione interna: .00110000 mantissa e 00000101 esponente

differenza 22.75 - 22: .10110110  
                          .10110000  
                          -----  
                          .00000110 mantissa e 00000101 esponente

ovvero: 0.110 che in decimale vale  $0.5 + 0.25 = 0.75$

### Esercizio 2

Il programma inizializza una matrice M di N righe e N colonne (5x5 elementi) contenente numeri interi. Viene eseguito quindi un ciclo (while) per i valori di k che variano fra 1 e 3 (<N-1), all'interno del quale viene chiamata la funzione F con parametri la matrice M e il valore k.

Al termine del ciclo, il programma stampa il contenuto della matrice modificata dalla funzione F. La funzione F prende come parametro lo stesso valore k per indicare rispettivamente un indice di riga e un indice di colonna per individuare un elemento della matrice M da modificare. In particolare saranno modificati i soli valori sulla diagonale principale.

Iterazioni:

k = 1.  $M[1][1] = M[1][1] + M[1][0]*M[0][1] = 3 + 4*4 = 19$

k = 2.  $M[2][2] = M[2][2] + M[2][1]*M[1][2] = 1 + 2*2 = 5$

k = 3.  $M[3][3] = M[3][3] + M[3][2]*M[2][3] = 4 + 5*5 = 29$

La matrice stampata sarà la seguente:

$M[0][0] = 5$	$M[0][1] = 4$	$M[0][2] = 3$	$M[0][3] = 2$	$M[0][4] = 1$
$M[1][0] = 4$	$M[1][1] = 19$	$M[1][2] = 2$	$M[1][3] = 1$	$M[1][4] = 5$
$M[2][0] = 3$	$M[2][1] = 2$	$M[2][2] = 5$	$M[2][3] = 5$	$M[2][4] = 4$
$M[3][0] = 2$	$M[3][1] = 1$	$M[3][2] = 5$	$M[3][3] = 29$	$M[3][4] = 3$
$M[4][0] = 1$	$M[4][1] = 5$	$M[4][2] = 4$	$M[4][3] = 3$	$M[4][4] = 2$

### Esercizio 3

```
#include<stdio.h>
#define MAX_CORSI 50
#define MAX_ORARI 3

typedef struct{
    char giorno[10];
    int oraInizio;
    int oraFine;
}orario;

typedef struct{
    int codice;
    char tipologia[15];
    int numMaxPartecipanti;
    int numPartecipanti;
    int numOrari;
    orario orari[MAX_ORARI];
}corso;

typedef enum {false, true} boolean;

void InserisciCorso(corso V[], corso c, int *numCorsi);
void CorsiDisponibiliPerTipologia(corso V[], int num, char tipo[]);

main()
{

boolean finito = false;
int numOrari, numCorsi=0, i;
char scelta, tipologia[15];
orario o;
corso c, CORSI[MAX_CORSI];

do{
    printf("Inserire il codice del corso: ");
    scanf("%d", &c.codice);

    printf("Inserire la tipologia del corso (principianti, intermedio,
avanzato): ");
    scanf("%s", c.tipologia);

    printf("Inserire il numero massimo di partecipanti: ");
    scanf("%d", &c.numMaxPartecipanti);

    c.numPartecipanti = 0;

    printf("Quanti sono gli orari del corso? (max %d) ",MAX_ORARI);
    scanf("%d", &c.numOrari);

    for(i=0;i<c.numOrari;i++){
        printf("Inserire il giorno numero %d: ", i+1);
        scanf("%s", o.giorno);

        printf("Inserire l'ora di inizio: ");
        scanf("%d", &o.oraInizio);

        printf("Inserire l'ora di fine: ");
        scanf("%d", &o.oraFine);

        c.orari[i] = o;
    }
}
}
```

```

    }

    InserisciCorso(CORSI,c,&numCorsi);

    printf("Vuoi continuare? (s/n)");
    scanf("%c", &scelta); /* per pulire il buffer di input */
    scanf("%c", &scelta);
    if (scelta == 'n' || scelta == 'N') finito = true;
}while(!finito && numCorsi < MAX_CORSI);

/* qui avviene l'iscrizione di partecipanti ai corsi.
   Fase non richiesta dal testo */

printf("Inserire la tipologia di corso da ricercare (principianti, intermedio,
avanzato): ");
scanf("%s", tipologia);

printf("Tipologia scelta: %s\n",tipologia);
printf("\n");
CorsiDisponibiliPerTipologia(CORSI,numCorsi,tipologia);
}

void InserisciCorso(corso V[], corso c, int *numCorsi){
    if(*numCorsi < MAX_CORSI) V>(*numCorsi++) = c;
    else printf("Vettore dei corsi pieno!");
}

void CorsiDisponibiliPerTipologia(corso V[], int num, char tipo[12]){
int i,j,posti;
for(i=0; i<num; i++){
    if(!strcmp(V[i].tipologia,tipo) && (posti=V[i].numMaxPartecipanti -
V[i].numPartecipanti) > 0){
        printf("Codice corso: %d\n", V[i].codice);
        for(j=0;j< V[i].numOrari; j++){
            printf("Orario: %s ", V[i].orari[j].giorno);
            printf("%d-%d\n", V[i].orari[j].oraInizio,
V[i].orari[j].oraFine);
        }
        printf("Numero di posti disponibili: %d\n", posti);
        printf("\n");
    }
}
}
}

```