



Programmazione I



dott. Sabrina Senatore
Dipartimento di Matematica e Informatica

Esercizio

Scrivere un programma che presa in input una data con il formato specificando gg mm aaaa, mediante l'impiego di funzioni:

- controlli se l'anno aaaa è bisestile;
- calcoli i giorni passati dall'inizio dell'anno.

Esempio:

Input: 12 12 2005

Output:

“IL 2005 non è bisestile”

“I giorni trascorsi dall'inizio dell'anno sono: 346

P.s. Si ricordi che un anno per essere bisestile deve essere divisibile per 4 ma non per 100 a meno che sia divisibile per 400. Ad esempio il 2004 è un anno bisestile così come il 2000 mentre non lo è il 1900 Si usi l'operazione modulo % che restituisce il resto di una divisione

```
#include <stdio.h>
int isBisestile(int);
int main()
{
    int arr_ggXmese[]={0,31,28,31,30,31,30,31,31,30,31,30,31};
    int giorno, mese, anno, totaleGG=0, i;

    printf("dammi la data gg-mm-aaaa: ");
    scanf("%d%d%d", &giorno, &mese, &anno);
        if (isBisestile(anno)==1)
            printf("L'anno %d risulta bisestile\n", anno);

        else
            printf("L'anno %d non risulta bisestile\n", anno);

    for ( i=1; i<mese; i++)
        totaleGG=totaleGG+arr_ggXmese[i];
    if (isBisestile(anno)==1 && mese>2)
        totaleGG++;
    printf("I giorni trascorsi dall'inizio dell'anno sono: %d\n", totaleGG+giorno);
    return 0;
}

int isBisestile(int anno){
    int bisestile=0;
    if (anno%4==0 && anno%100!=0 || anno%400==0)
        bisestile=1;
    return bisestile;
}
```

```
#include <stdio.h>
int isBisestile(int);
int calcoloGG(int, int);
int main()
{
    int giorno, mese, anno, i, gg;
    printf("dammi la data gg-mm-aaaa: ");
    scanf("%d%d%d", &giorno, &mese, &anno);
    if (isBisestile(anno)==1)
        printf("L'anno %d risulta bisestile\n", anno);
    else
        printf("L'anno %d non risulta bisestile\n", anno);
    gg= calcoloGG(mese, giorno);
    if (isBisestile(anno)==1 && mese>2)
        gg++;
    printf("I giorni trascorsi dall'inizio dell'anno sono: %d\n", gg);
    return 0;
}
int isBisestile(int anno)
{
    int bisestile=0;
    if (anno%4==0 && anno%100!=0 || anno%400==0)
        bisestile=1;
    return bisestile;
}
int calcoloGG(int mese, int giorno)
{
    int totaleGG=0;
    int i;
    int arr_ggXmese[]={0,31,28,31,30,31,30,31,31,30,31,30,31};
    for ( i=1; i<mese; i++)
        totaleGG=totaleGG+arr_ggXmese[i];
    totaleGG=totaleGG+giorno;
    return totaleGG;
}
```

Esercizio

- Una sequenza di 27 interi si dice perfetta se consiste di tre 1, tre 2, . . . , tre 9, posizionati in modo che per ogni $i \in [1..9]$ ci sono esattamente i numeri tra le occorrenze successive di i .
- Ad esempio, la sequenza
- (1,9,1,2,1,8,2,4,6,2,7,9,4,5,8,6,3,4,7,5,3,9,6,8,3,5,7) è perfetta.
- Si scriva una funzione C che prende come parametro un vettore di 27 interi e restituisce 1 se il vettore contiene una sequenza perfetta e 0 altrimenti.

```
#include <stdio.h>
int perfetta(int [], int);
int main()
{
    int arr[]={1,9,1,2,1,8,2,4,6,2,7,9,4,5,8,6,3,4,7,5,3,9,6,8,3,5,7};
    if ( perfetta(arr, 27))
        printf("sequenza perfetta");
    else
        printf("sequenza non perfetta");
    return 0;
}
int perfetta(int v[], int n){
    int num, j;
    for (num=1; num<9; num++) {
        for (j = 0; j <= 24- 2*num; j++)
            if (v[j] == num && v[j+num+1] == num && v[j+2*num+2] == num)
                break;
        if (j > 24-2*num)
            return 0;
    }
    return 1;
}
```

Esercizio

- Sia dato un vettore di caratteri presente in memoria. Si scriva una funzione C che ricevendo, in ingresso il vettore restituisca, tramite passaggio di parametri, il carattere che più frequentemente degli altri è seguito dal carattere successivo nell'ordine alfabetico.
- Ad esempio, se il vettore contiene i caratteri
A F L M P S T L M
la funzione dovrà restituire il carattere L, che per due volte è seguito dal carattere M.

7