

## Relazione per “IlCalendario”

Si cominci con l’illustrazione della descrizione del progetto:

|                                |  |                |
|--------------------------------|--|----------------|
| <b>Titolo Progetto:</b>        | IlCalendario   |                |
| <b>Nome del programmatore:</b> | Giuseppe Carlino   |                |
| <b>Specifiche tecniche:</b>    | Linguaggio (con versione):   | C99            |
|                                | IDE (con versione):  | CLion 2023.2.2 |
| <b>Descrizione:</b>            | IlCalendario è un software finalizzato a riprodurre un semplicissimo calendario utilizzando la CLI. Basta inserire l'anno in input e il gioco è fatto. |                |

Si mostri anche la guida dell’applicazione:

## Guida per “IlCalendario”

Innanzitutto, dopo aver lanciato l’applicazione, viene illustrato lo scopo del software e si invita l’utente ad inserire un anno;

```
C:\Users\39327\Documents\Progetti\IlCalendario\IlCalendario\cmake-build-debug\IlCalendario.exe
+++++IlCalendario+++++
-> Lo scopo del software e' quello di stampare a video il calendario dell'anno inserito in input dall'utente.
-> Inserire l'anno di cui si vuole conoscere il calendario (min: 1582; max: 9999):
```

Nel caso non venisse rispettato l'intervallo indicato tra parentesi, verrebbe fatto visualizzare un messaggio di errore seguito da un nuovo invito all'inserimento di un anno;

Inserire l'anno di cui si vuole conoscere il calendario (min: 1582; max: 9999): 1581  
Attenzione! Rispettare l'intervallo indicato.

Dopo aver fornito l'input corretto, viene stampato a video il calendario dell'anno scelto accompagnato da un "bing".

```
Inserire l'anno di cui si vuole conoscere il calendario (min: 1582; max: 9999): 2024

*****2024*****

-----
          Gennaio
L   M   M   G   V   S   D
1   2   3   4   5   6   7
8   9  10  11  12  13  14
15  16  17  18  19  20  21
22  23  24  25  26  27  28
29  30  31
-----
          Febbraio
L   M   M   G   V   S   D
    5   6   7   8   9  10  11
12  13  14  15  16  17  18
19  20  21  22  23  24  25
26  27  28  29
-----
          Marzo
```

Ecco riportato il dizionario dati del software.

| Dizionario dati:    |         |              |   |
|---------------------|---------|--------------|---|
| -----               |         |              |   |
| Nome costante       | Tipo    | Utilizzo     | Descrizione   |
| -----               |         |              |   |
| NOME_MESI[12][10]   | Stringa | Lavoro       | Matrice contenente il nome di tutti i mesi dell'anno.   |
| MAX_GIORNI_MESI[12] | Intero  | Lavoro       | Vettore contenente il numero di giorni di tutti i mesi dell'anno.   |
| -----               |         |              |   |
| Nome variabile      | Tipo    | Utilizzo     | Descrizione   |
| -----               |         |              |   |
| anno_scelto         | Intero  | Input/Output | Anno di cui l'utente desidera visualizzare il calendario.   |
| anno_rif            | Intero  | Lavoro       | Anno da prendere come riferimento per la costruzione del calendario dell'anno anno_scelto.  |
| dif_anni            | Intero  | Lavoro       | Differenza di anni tra anno_rif e anno_scelto.  |
| primo_giorno        | Intero  | Lavoro       | Primo giorno di ogni mese dell'anno.  |
| mese                | Intero  | Output       | Indice delle righe della matrice NOME_MESI[12][10]. Indica il mese da stampare a video.   |
| vuoto               | Intero  | Lavoro       | Indica i vuoti che ci sono prima del primo giorno di ogni mese dell'anno.   |
| giorno              | Intero  | Output       | Indica il giorno di quel mese da stampare a video. Si prolunga fino al valore numerico all'indice mese del vettore MAX_GIORNI_MESI[12]. |
| flag_bisestile      | Intero  | Lavoro       | Flag utile a verificare se un anno e' bisestile o no.   |
| -----               |         |              |   |

Si illustrino le librerie utilizzate:

- **stdio.h** (da “**Standard input output**”): fornisce le funzioni/procedure per compiere le operazioni basiche di input/output;
- **stdlib.h** (da “**Standard library**”): contiene qualsiasi delle funzioni/procedure di utilità generale (es.: operazioni di allocazione di memoria).

```
// Inclusione librerie utilizzate
#include <stdio.h> // Includere la libreria standard STANDARD INPUT/OUTPUT
#include <stdlib.h> // Includere la libreria standard STANDARD LIBRARY
```

Si mostrino le costanti di cui si è fatto uso:

```
// Dichiarazione ed implementazione costanti
const char NAME_MESI[12][10] = { (0) "Gennaio", (1) "Febbraio", (2) "Marzo", (3) "Aprile", (4) "Maggio", (5) "Giugno", (6) "Luglio", (7) "Agosto", (8) "Settembre", (9) "Ottobre", (10) "Novembre", (11) "Dicembre"}; // Nome dei 12 mesi dell'anno
const int MAX_GIORNI_MESI[12] = { (0) 31, (1) 29, (2) 31, (3) 30, (4) 31, (5) 30, (6) 31, (7) 31, (8) 30, (9) 31, (10) 30, (11) 31}; // Numero massimo di giorni di ogni mese dell'anno (ad eccezione di febbraio, che non e' sempre costituito da 29 giorni)
```

Questi sono tutti i prototipi di funzioni/procedure create per facilitare la programmazione. Lo scopo di ognuna di queste è illustrato in basso.

```
// Dichiarazione prototipi di funzioni e/o procedure
↪ int primo_giorno_inizio(int);
↪ int primo_giorno_fine(int);
↪ int controllo_bisestile(int);
↪ void calcolo_primo_giorno(int);
↪ void stampa_calendario(int, int);
```

Si termini con la descrizione del codice sorgente del software:

**int main():**

1. Si comincia con la dichiarazione di variabili locali;

```
int main() {

    // Dichiarazione variabili
    int anno_scelto;
```

2. Dopodiché, tramite l'utilizzo della funzione system della libreria stdlib.h, si cambiano i colori di background e foreground della finestra del software (sfondo azzurro indicato dal “9” e testo bianco indicato da “F” in questo caso);

```
// Assegnazione di un colore allo sfondo e al testo della CLI del software
system( Command: "COLOR 9F");
```

3. Qui vengono stampati il titolo e lo scopo dell'app;

```
// Titolo del software
printf("+++++IlCalendario+++++\n\n");

// Istruzioni finalizzate a garantire il corretto utilizzo del software
printf("Lo scopo del software e' quello di stampare a video il calendario dell'anno inserito in input dall'utente.\n\n\n");
```

4. Successivamente, si chiede all'utente di inserire l'anno di cui preferisce sapere il calendario. Aspetto fondamentale è rispettare l'intervallo indicato tra parentesi;

```
// Inserimento dell'anno di cui si vuole conoscere il calendario
do {
    printf("Inserire l'anno di cui si vuole conoscere il calendario (min: 1582; max: 9999): ");
    scanf("format: %d", &anno_scelto);

    if (anno_scelto < 1582 || anno_scelto > 9999)
        printf("Attenzione! Rispettare l'intervallo indicato.\n\n");
} while (anno_scelto < 1582 || anno_scelto > 9999);
```

5. Poi si procede con il richiamare la procedura calcolo\_primo\_giorno;

```
// Calcolo del primo giorno dell'anno scelto (L, M, M, G, V, S o D)
calcolo_primo_giorno( anno_scelto);
```

6. Infine, il gioco va in pausa (a causa del valore "PAUSE" fornito come parametro alla funzione system) e si chiude con la pressione di un tasto.

```
// Mettere in pausa il software alla fine della sua esecuzione
system( Command: "PAUSE");

// Chiudere la funzione main
return 0;
}
```

**int primo\_giorno\_inizio(int pprimo\_giorno):**

1. Si comincia con la stesura dei parametri della funzione. Il parametro intero pprimo\_giorno sta ad indicare un giorno della settimana;

```
// Funzione finalizzata a riportare a 1 il valore di primo_giorno se questo supera il valore 7
int primo_giorno_inizio(int pprimo_giorno) {
```

2. La verifica è semplice: se il parametro supera il valore 7, la funzione restituisce il valore 1 per ricominciare la numerazione da lunedì, altrimenti restituisce il valore del parametro stesso.

```
// Reset di primo_giorno a 1
if (pprimo_giorno > 7)
    return 1;

// Nessun reset
else
    return pprimo_giorno;
}
```

**int primo\_giorno\_fine(int pprimo\_giorno):**

1. Si comincia con la stesura dei parametri della funzione. Il parametro intero pprimo\_giorno assume la stessa funzione attribuitagli in primo\_giorno\_inizio;

```
// Funzione finalizzata a riportare a 7 il valore di primo_giorno se questo scende sotto il valore 1
int primo_giorno_fine(int pprimo_giorno) {
```

2. La verifica è semplice: se il parametro scende sotto il valore 1, la funzione restituisce il valore 7 per ricominciare la numerazione da domenica, altrimenti restituisce il valore del parametro stesso.

```
// Reset di primo_giorno a 7
if (pprimo_giorno < 1)
    return 7;

// Nessun reset
else
    return pprimo_giorno;
}
```

**int controllo\_bisestile(int panno):**

1. Si comincia con la stesura dei parametri della funzione. Il parametro intero panno rappresenta l'anno da verificare (capire se è bisestile o no);

```
// Funzione finalizzata a verificare se un anno e' bisestile o no
int controllo_bisestile(int panno) {
```

2. La verifica è semplice: l'anno inserito dall'utente è bisestile se: l'anno è secolare ed è divisibile per 4 e per 400; l'anno non è secolare ma è divisibile per 4.

```
// Anno secolare
if (panno % 100 == 0) {
    if (panno % 4 == 0 && pann % 400 == 0)
        return 1;
    else
        return 0;

// Anno non secolare
} else if (panno % 4 == 0) {
    return 1;

// Anno non bisestile
} else {
    return 0;
}
}
```

**void calcolo\_primo\_giorno(int panno\_scelto):**

1. Si comincia con la stesura dei parametri della funzione e con la dichiarazione di variabili locali. Il parametro panno\_scelto rappresenta l'anno inserito in input dall'utente;

```
// Procedura finalizzata a calcolare il primo giorno dell'anno scelto
void calcolo_primo_giorno(int panno_scelto) {

    // Dichiarazione (ed implementazione) variabili
    int anno_rif = 4210, dif_anni, primo_giorno = 1, flag_bisestile;
```

2. Si procede con il calcolo di dif\_anni;

```
// Calcolo della differenza di anni tra anno_rif e panno_scelto
dif_anni = anno_rif - panno_scelto;
```



3. Se `dif_anni` risulta essere maggiore di 0, si fa scorrere `anno_rif` (che inizia di lunedì, primo giorno della settimana, ed è più o meno a metà dei valori estremi all'intervallo posto nell'input dell'utente, per eguagliare i tempi di calcolo nel caso l'anno scelto fosse inferiore o superiore al 4210) all'indietro fino al valore di `panno_scelto` e si decrementa di 1 `primo_giorno` ad ogni ciclo (di 2 se l'anno precedente a quello rappresentato da `anno_rif` è bisestile. Per verificarlo si richiama la funzione `controllo_bisestile`. Infatti l'anno successivo ad uno bisestile trasla di 2 il suo primo giorno rispetto al precedente), richiamando ogni volta la funzione di controllo `primo_giorno_fine`;

```
// Se la differenza e' maggiore di 0...
if (dif_anni > 0) {

    // Scorrere anno_rif all'indietro fino ad anno_scelto
    for (anno_rif = 4210; anno_rif > panno_scelto; anno_rif--) {

        // Decrementare di 1 primo_giorno
        primo_giorno--;

        // Verificare che primo_giorno sia sceso sotto il valore di 1
        primo_giorno = primo_giorno_fine( pprimo_giorno: primo_giorno);

        // Verificare che l'anno indicato da anno_rif - 1 sia bisestile o no
        flag_bisestile = controllo_bisestile( panno: anno_rif - 1);

        if (flag_bisestile) {

            // Decrementare ancora di 1 primo_giorno se anno_rif - 1 e' bisestile
            primo_giorno--;

            // Quindi verificare di nuovo che primo_giorno sia sceso sotto il valore di 1
            primo_giorno = primo_giorno_fine( pprimo_giorno: primo_giorno);

        }

    }

}
```

4. Se invece dif\_anni risulta essere minore di 0, si fa più o meno il contrario di quanto visto precedentemente. Ovviamente, se dif\_anni risulta essere proprio uguale a 0 (nel caso l'utente volesse conoscere il calendario del 4210), non viene eseguito nessuno dei due cicli;

```
} else {  
  
    // Se la differenza e' minore di 0...  
    if (dif_anni < 0) {  
  
        // Scorrere anno_rif in avanti fino ad anno_scelto  
        for (anno_rif = 4210; anno_rif < panno_scelto; anno_rif++) {  
  
            // Incrementare di 1 primo_giorno  
            primo_giorno++;  
  
            // Verificare che primo_giorno abbia superato il valore 7  
            primo_giorno = primo_giorno_inizio( pprimo_giorno: primo_giorno);  
  
            // Verificare che l'anno indicato da anno_rif sia bisestile o no  
            flag_bisestile = controllo_bisestile( panno: anno_rif);  
  
            if (flag_bisestile) {  
  
                // Incrementare ancora di 1 primo_giorno se anno_rif e' bisestile  
                primo_giorno++;  
  
                // Quindi verificare di nuovo che primo_giorno abbia superato il valore 7  
                primo_giorno = primo_giorno_inizio( pprimo_giorno: primo_giorno);  
            }  
        }  
    }  
}
```

5. Una volta calcolato il primo giorno dell'anno, si può finalmente stampare il calendario dell'anno selezionato dall'utente richiamando la procedura stampa\_calendario con anche primo\_giorno dato come parametro di input.

```
printf("\n\n");  
  
// Stampa del calendario relativo all'anno scelto  
stampa_calendario(panno_scelto, pprimo_giorno: primo_giorno);  
}
```



**void stampa\_calendario(int panno\_scelto, int pprimo\_giorno):**

1. Si comincia con la stesura dei parametri della funzione e con la dichiarazione di variabili locali. i parametri panno\_scelto e pprimo\_giorno assumono le stesse funzioni indicate nei casi precedenti;

```
// Procedura finalizzata a stampare il calendario dell'anno scelto
void stampa_calendario(int panno_scelto, int pprimo_giorno) {

    // Dichiarazione variabili
    int mese, vuoto, giorno, flag_bisestile;
```

2. Dopodiché, verificare che panno\_scelto sia bisestile (assegnando il valore restituito da quest'ultima a flag\_bisestile) mediante il richiamo della funzione controllo\_bisestile;

```
// Verificare che l'anno scelto sia bisestile o no
flag_bisestile = controllo_bisestile( panno: panno_scelto);
```

3. Andare avanti con lo stampare a video l'anno scelto dall'utente come titolo del calendario;

```
// Stampa dell'anno scelto
printf("\a*****%d*****\n", panno_scelto);
```

4. Iniziare poi con la scrittura a video dei vari mesi dell'anno. Per ogni mese del calendario, stampare una barra separatoria e una barra dei giorni della settimana sottostante al nome del mese successivo. Lasciare poi tante caselle vuote quant'è il valore di pprimo\_giorno, in modo da stampare correttamente (in fase con il mese di prima) il primo giorno del mese;

```
// Stampa dei mesi dell'anno
for (mese = 0; mese < 12; mese++) {

    // Stampa di una barra separatoria per ognuno dei mesi dell'anno
    printf("\n-----\n");

    // Stampa del nome del mese e della riga indicante i giorni della settimana
    printf("          %s\n", NOME_MESI[mese]);
    printf("L    M    M    G    V    S    D\n");

    // Verificare che primo_giorno abbia superato il valore 7
    pprimo_giorno = primo_giorno_inizio(pprimo_giorno);

    // Stampa dei vuoti presenti prima del primo giorno del mese
    for (vuoto = 1; vuoto < pprimo_giorno; vuoto++)
        printf("    ");
```

5. Per ognuno dei mesi dell'anno, stampare a video tutti i giorni che li compongono. Andare a capo per ogni settimana trascorsa (cioè quando pprimo\_giorno, utile a trovare il primo giorno di ogni mese, supera il valore 7. Tale valore viene quindi riportato a 1), eccetto se l'anno non è bisestile e capita il 29 di febbraio nella conta dei giorni, dato che non esiste tale giorno in un anno non bisestile. Incrementare poi pprimo\_giorno per ogni giorno trascorso;

```
// Stampa dei giorni del mese
for (giorno = 1; giorno <= MAX_GIORNI_MESI[mese]; giorno++) {

    // Se passa una settimana...
    if (pprimo_giorno > 7) {

        // Andare a capo ad ogni settimana passata
        // Se capita il giorno 29 di febbraio, farlo solo se l'anno associato e' bisestile
        if (giorno == 29 && mese == 1) {
            if (flag_bisestile)
                printf("\n");
        } else {
            printf("\n");
        }

        // Reset di primo_giorno a 1
        pprimo_giorno = 1;
    }

    // Incrementare di 1 primo_giorno
    pprimo_giorno++;
}
```

6. Se il giorno è costituito da 1 cifra, stamparlo seguito da 4 spazi, altrimenti stamparne solo 3 se è costituito da 2 cifre. Se l'anno non è bisestile, non bisogna stampare il 29 di febbraio: quindi decrementare di 1 pprimo\_giorno e verificare che non debba ripartire da 7;

```
// Se il giorno e' costituito da una cifra...
if (giorno < 10) {

    // Stampare il giorno ed uno spazio in piu'
    printf("%d   ", giorno);

// Se cosi' non e'...
} else {

    // Se capita il giorno 29 di febbraio...
    if (giorno == 29 && mese == 1) {

        // Stampare il suddetto giorno solo se l'anno e' bisestile
        if (flag_bisestile) {
            printf("%d", giorno);

            // In caso contrario...
        } else {

            // Decrementare di 1 primo_giorno
            pprimo_giorno--;

            // Verificare che primo_giorno sia sceso sotto il valore di 1
            pprimo_giorno = primo_giorno_fine(pprimo_giorno);

        }

        // Altrimenti...
    } else {

        // Stampa di un giorno del mese
        printf("%d   ", giorno);

    }

}

}
```

7. Infine, stampare una barra separatoria utile a chiudere il calendario.

```
// Stampa dell'ultima barra separatoria del calendario
printf("\n-----\n\n\n");

}
```