

Estructura del Programa de Calendario en C++

Estructura General del Programa:

El programa está diseñado para mostrar un calendario de un año específico ingresado por el usuario. Utiliza varias funciones para organizar el código y realizar tareas específicas, como verificar si un año es bisiesto, obtener el número de días en un mes, imprimir un mes y finalmente imprimir todo el año.

Paso a Paso del Código:

Inclusión de Bibliotecas:

```
#include <iostream>

#include <string> // Incluir la biblioteca de string

using namespace std;
```

Se incluyen las bibliotecas necesarias. `iostream` permite la entrada y salida estándar (como `cin` y `cout`), y `string` permite el uso de la clase `string` para manejar cadenas de texto.

Función `esAñoBisiesto(int año)`:

```
bool esAñoBisiesto(int año) {

    return (año % 4 == 0 && año % 100 != 0) || (año % 400 == 0);

}
```

Propósito: Determina si un año es bisiesto.

Lógica: Un año es bisiesto si:

- Es divisible por 4.

Estructura del Programa de Calendario en C++

- No es divisible por 100, a menos que también sea divisible por 400.

Retorno: Devuelve true si el año es bisiesto y false en caso contrario.

Función verDiasUnMes(int mes, int año):

```
int verDiasUnMes(int mes, int año) {  
    int diasUnMes[] = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };  
    if (mes == 2 && esAñoBisiesto(año)) {  
        return 29;  
    }  
    return diasUnMes[mes - 1];  
}
```

Propósito: Devuelve el número de días en un mes específico de un año dado.

Lógica:

- Se utiliza un arreglo diasUnMes que contiene el número de días para cada mes (de enero a diciembre).
- Si el mes es febrero (mes 2) y el año es bisiesto, devuelve 29 días; de lo contrario, devuelve el número de días correspondiente al mes.

Retorno: El número de días en el mes especificado.

Función mostrarMes(int mes, int año):

```
void mostrarMes(int mes, int año) {
```

Estructura del Programa de Calendario en C++

```
string nombresMeses[] = {  
    'Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio',  
    'Julio', 'Agosto', 'Septiembre', 'Octubre', 'Noviembre', 'Diciembre'  
};  
  
cout << '    ' << nombresMeses[mes - 1] << ' ' << año << endl;  
cout << 'Do Lu Ma Mi Ju Vi Sa' << endl;  
  
int primerDia = (1 + (año - 1) + (año - 1) / 4 - (año - 1) / 100 + (año - 1) / 400) % 7;  
for (int m = 1; m < mes; m++) {  
    primerDia = (primerDia + verDiasUnMes(m, año)) % 7;  
}  
  
for (int i = 0; i < primerDia; i++) {  
    cout << ' '; // Imprimir espacios para los días antes del primer día del mes  
}  
  
int diasUnMes = verDiasUnMes(mes, año);  
for (int dia = 1; dia <= diasUnMes; dia++) {  
    if (dia < 10) {  
        cout << ' ' << dia << ' '; // Añadir un espacio para días de un solo dígito  
    } else {  
        cout << dia << ' '; // Imprimir días de dos dígitos normalmente  
    }  
    primerDia++; // Pasar al siguiente día
```

Estructura del Programa de Calendario en C++

```
if (primerDia % 7 == 0) {  
    cout << endl; // Comenzar una nueva línea después del sábado  
}  
  
}  
  
cout << endl; // Imprimir una nueva línea después de que el mes haya terminado  
cout << endl; // Imprimir una nueva línea después de que el mes haya terminado  
}
```

Propósito: Imprime el calendario de un mes específico.

Lógica:

- Se define un arreglo nombresMeses que contiene los nombres de los meses.
- Se imprime el nombre del mes y el año.
- Se calcula el primer día del mes utilizando una fórmula que considera el año y los días transcurridos hasta el mes actual.
- Se imprimen espacios en blanco para alinear correctamente los días del mes según el día de la semana en que comienza.
- Se imprimen los días del mes, asegurando que se mantenga el formato adecuado (con un espacio adicional para los días de un solo dígito).

Salida: Muestra el calendario del mes en la consola.

Función mostrarAño(int año):

```
void mostrarAño(int año) {  
    for (int mes = 1; mes <= 12; mes++) {
```

Estructura del Programa de Calendario en C++

```
    mostrarMes(mes, año); // Imprimir cada mes del año
}
}
```

Propósito: Imprime el calendario completo de un año.

Lógica: Llama a la función mostrarMes para cada mes del año (de enero a diciembre).

Salida: Muestra el calendario de todos los meses del año en la consola.

Función main():

```
int main() {
    cout << 'Este es un programa que muestra el calendario de un año específico.' << endl;
    cout << endl;

    int año;

    cout << 'Ingrese el año: ';
    cin >> año; // Obtener el año del usuario

    mostrarAño(año); // Imprimir el calendario para el año especificado
    return 0; // Indicar que el programa terminó con éxito
}
```

Propósito: Punto de entrada del programa.

Estructura del Programa de Calendario en C++

Lógica:

- Se muestra un mensaje introductorio al usuario.
- Se solicita al usuario que ingrese un año.
- Se llama a mostrarAño con el año ingresado para mostrar el calendario completo.

Salida: El programa termina devolviendo 0, indicando que se ejecutó correctamente.

Resumen:

Este programa es un buen ejemplo de cómo se pueden utilizar funciones en C++ para organizar el código y realizar tareas específicas de manera modular. Además, demuestra el uso de estructuras de control, arreglos y la manipulación de cadenas para crear una aplicación útil y visualmente clara. Al final, el usuario puede ver el calendario del año que ingresó, con los días correctamente alineados según el día de la semana.