



Problema 1.º

Diseña el código de la función `diasDelMes` cuya especificación se presenta a continuación:

```
/*  
 * Pre: 1 ≤ mes ≤ 12 y agno > 1582.  
 * Post: Devuelve el número de días que tiene el mes establecido por el parámetro  
 *       «mes» del año establecido por el parámetro «agno».  
 */  
int diasDelMes (int mes, int agno) {  
    ...  
}
```

Puedes considerar que la función `esBisiesto` de la clase de problemas anterior ya está disponible:

```
/*  
 * Pre: agno > 1582.  
 * Post: Ha devuelto true si y solo si el año «agno» es bisiesto de acuerdo con el  
 *       calendario gregoriano.  
 */  
bool esBisiesto(int agno);
```

Problema 2.º

Utiliza la función anterior para escribir un programa que tenga el siguiente comportamiento iterativo:

```
Escriba un mes y un año: 10 2019  
Este mes tiene 31 días.  
  
Escriba un mes y un año: 11 2019  
Este mes tiene 30 días.  
  
Escriba un mes y un año: 2 2019  
Este mes tiene 28 días.  
  
Escriba un mes y un año: 2 2019  
Este mes tiene 28 días.  
  
Escriba un mes y un año: 2 2020  
Este mes tiene 29 días.  
  
Escriba un mes y un año: 21 2019  
El mes tiene que estar comprendido entre 1 y 12.  
  
Escriba un mes y un año: 0 1992  
El mes tiene que estar comprendido entre 1 y 12.  
  
Escriba un mes y un año: -6 2008  
El mes tiene que estar comprendido entre 1 y 12.  
  
Escriba un mes y un año: 10 1492  
El año tiene que ser posterior a 1582.  
  
Escriba un mes y un año: 0 0
```

Como se puede comprobar en el ejemplo de ejecución, el programa pregunta repetidamente por un mes y un año. Si el mes está comprendido entre 1 y 12 y el año es posterior a 1582, el programa escribe en la pantalla el número de días que tiene el mes. En caso contrario, escribe un mensaje de error. El programa termina cuando el usuario escribe 0 tanto para el mes como para el año.



Problema 3.º

Diseña el código de una función denominada `diaSiguiente`, en la que puedes (y se recomienda), utilizar la función `diasDelMes` del problema 1º. En el grupo de mañanas, el paso de parámetros por referencia se explicará el viernes 18 de octubre. Por eso, para la clase de problemas del 17 de octubre, en el caso del grupo de mañanas se pide diseñar una versión diferente de la función `diaSiguiente` a la del grupo de tardes.

Grupo de mañanas

```
/*
 * Pre: 1 ≤ dia ≤ 31, 1 ≤ mes ≤ 12, agno > 1582 y la fecha formada por
 * «dia», «mes» y «agno» representan una fecha válida del calendario
 * gregoriano.
 * Post: Ha devuelto la fecha correspondiente al día posterior a la fecha
 * representada por el valor de los parámetros «dia», «mes» y «agno»
 * codificada con el mismo formato «aaaammdd» utilizado en la práctica 2ª,
 * donde los dígitos «aaaa» representan el año, los dígitos «mm», el mes y los
 * dígitos «dd» el día.
 *
 * Por ejemplo:
 * diaSiguiente(17, 10, 2019) devuelve 20191018
 * diaSiguiente(29, 2, 2020) devuelve 20200301
 * diaSiguiente(31, 12, 2022) devuelve 20230101
 */
int diaSiguiente(int dia, int mes, int agno) {
    ...
}
```

Grupo de tardes

```
/*
 * Pre: 1 ≤ dia ≤ 31, 1 ≤ mes ≤ 12, agno > 1582 y la fecha formada por
 * «dia», «mes» y «agno» representan una fecha válida del calendario
 * gregoriano.
 * Post: Tras la ejecución de la función, los parámetros «fecha», «dia», «mes» y
 * «agno» representan la fecha correspondiente al día siguiente al que
 * representaban al iniciarse la ejecución de la función.
 *
 * Por ejemplo, si d, m y a son variables de tipo entero y d = 17, m = 10 y
 * a = 2019, tras la invocación diaSiguiente(d, m, a) los valores de las
 * variables serían d = 18, m = 10 y a = 2019.
 * Si los valores fueran d = 29, m = 2 y a = 2020, tras la invocación
 * diaSiguiente(d, m, a) los valores serían d = 1, m = 3 y a = 2020.
 * Si los valores fueran d = 31, m = 12 y a = 2022, tras la invocación
 * diaSiguiente(d, m, a) los valores serían d = 1, m = 1 y a = 2023.
 */
void diaSiguiente(int& dia, int& mes, int& agno) {
    ...
}
```

Problema 4.º

Escribe un programa completo que pida una fecha al usuario y escriba en pantalla la fecha correspondiente al día siguiente, utilizando la función `diaSiguiente` del problema anterior.