BOLETÍN DE EJERCICIOS TEMA 3

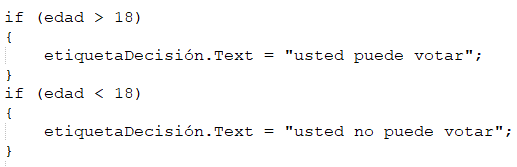
PROFESOR: MARTÍN GARCÍA FIGUEIRA

martin@ciclosmontecastelo.com

línea horizontal

1. **Comparación de sentencias**

¿Todas las opciones están cubiertas con las siguientes dos piezas de código de C#?

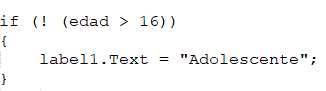


1. **Mostrando el valor mínimo**

Escribe un programa donde tengas un TrackBar y una caja de texto. Muestra en la caja de texto el valor numérico mínimo que haya tenido el TrackBar.

1. **Reescribiendo sentencias**

Reescribe este código sin utilizar la negación:



1. **Juego de dados**

Escribe un programa donde tengas dos TrackBar (cada una representa a un dado de 6 caras) y una caja de texto, que indica si pierdes o ganas.

Solamente se puede ganar si la suma de los dos dados es 6.

1. **¿Es válido?**

Escribe un programa con una caja de texto y una etiqueta. Implementa instrucciones if para evaluar si alguien puede obtener un empleo de tiempo completo. La regla es que debe tener 16 años o más, y ser menor de sesenta y cinco.

1. **Impuestos a pagar**

Escribe un programa que reciba como información de entrada una cantidad que represente un rango de salario; utiliza una TrackBar y determina qué cantidad de impuestos deben pagar los usuarios de acuerdo con las siguientes reglas:

* El usuario no tiene que pagar impuestos si gana hasta 10.000€
* Debe pagar 20% de impuestos si gana más de 10.000€ y hasta 50.000€
* Por último, debe pagar 90% de impuestos si gana más de 50.000€

La barra de seguimiento debe tener un rango de 0 a 100.000.

Utiliza una etiqueta o un MessageBox para mostrar qué cantidad de impuestos debe pagar un usuario.

1. **¿Cuál es el máximo?**

Escribe un programa que conste de tres TrackBar y muestre el mayor de los tres valores representados en ellas al hacer clic en un botón.

1. **Nombres de las cartas**

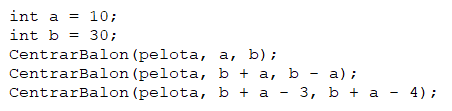
Escribe un método al que le pases un número entre 1-4 mediante una caja de texto y te devuelva su palo correspondiente (Oros-Bastos-Copas-Espadas) mediante un MessageBox.

1. **Llamada a métodos I**

Tenemos el siguiente método CentrarBalon:

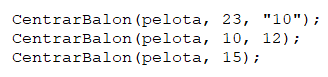


¿A qué posiciones llegarán los siguientes balones centrados?



1. **Llamada a métodos II**

¿Cuál es el error en estas llamadas?

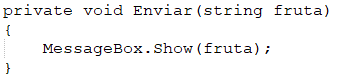


1. **Llamada a métodos III**

Tenemos esta llamada a un método:



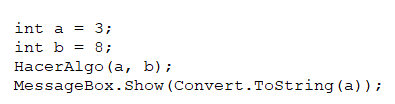
y éste es el código del método:



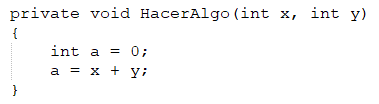
¿Qué ocurre cuando se invoca este método?

1. **Llamada a métodos IV**

Tenemos esta llamada a un método:



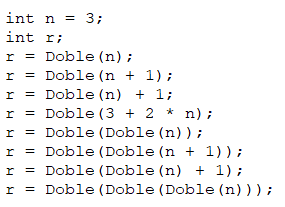
y éste es el código del método:



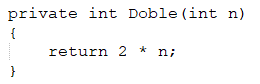
¿Qué se muestra en el mensaje?

1. **Llamada a métodos V**

Tenemos esta llamada a un método:



y éste es el código del método:



¿Cuál es el resultado de cada llamada?

1. **Cajero automático**

Imagina que un cajero automático sólo puede dispensar billetes de 10 y 5 €.

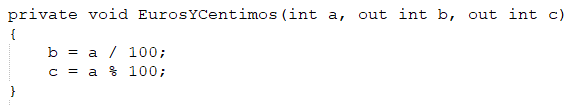
Escribe un método que calcule el número de billetes de 10 y 5 € para cualquier cantidad de dinero. Limita las cantidades a las que puedas dar con billetes de 10 y 5.

La definición del método debe tener esta forma:



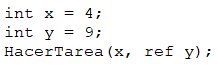
1. **Comprobación de método**

¿El funcionamiento de este método sería correcto?

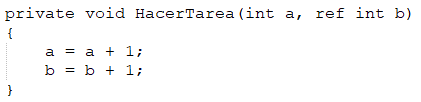


1. **Llamada a métodos VI**

Tenemos esta llamada a un método:



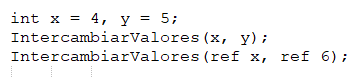
y éste es el código del método:



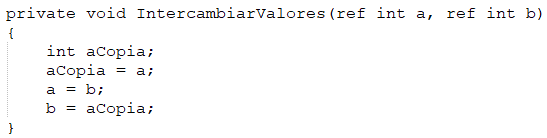
¿Cuáles son los valores resultantes de x e y?

1. **Llamada a métodos VII**

¿Hay algún error en las siguientes llamadas?



El código del método es:



1. **Calculadora sencilla**

Tendrá dos cajas de texto para los operandos y otra para mostrar el resultado. Deberá tener las funciones matemáticas básicas:

* 1. Coloca tres cajas de texto sobre el formulario, dos para los operandos y la tercera para el resultado.
  2. Añade cuatro botones, cambia el Text de los mismos a ‘+’, ‘-‘, ‘\*’ y ‘/’. Cambia el tamaño de la fuente hasta el adecuado. Añade un label ‘=’.
  3. Configura los botones de las operaciones Sumar y Restar

1. **Calculadora avanzada**

Añade código al programa de la Calculadora, para que multiplique y divida.

¿Crees que la división nos va a dar algún problema adicional?

1. **Ejercicio: Tabla del Dos**

* Inicia un nuevo proyecto y coloca un listbox sobre el formulario.
* Añade un botón con el texto ‘Tabla del dos’.
* Añade código correspondiente al Click del botón para mostrar la tabla del 2 por pantalla.
* Utilizando la misma estructura, muestra los primeros 10 factores de la tabla del 3
* Utilizando la misma estructura, muestra los primeros 10 factores de la tabla del 5, en orden descendente

1. **Limpiar cajas de texto**

* Coloca algunas cajas de texto y un botón sobre un formulario
* Escribe un método que limpie el texto en las cajas de texto.
* Este método no tendrá parámetros ni devolverá valor.
* Invoca a este método desde el Click del botón.

1. **Simulador de dado**

Crea una aplicación de escritorio donde tengas un método que no tenga parámetros y que retorne un número comprendido entre 1 y 6

1. **Cálculo del promedio**

Crea una aplicación de escritorio donde ingreses 5 números y muestren su promedio

1. **Cálculo del cuadrado**

Crea una aplicación de escritorio donde ingreses un número y muestres el cuadrado del mismo.

El número debe ser mayor que cero, en caso de error que aparezca el mensaje "ERROR. Vuelva a ingresar el número"

1. **Número par o impar**

Crea una aplicación de escritorio donde ingreses un número y te diga si es un número par o impar.

1. **Contar las vocales**

Crea una aplicación de escritorio donde ingreses una cadena y te cuente cuántas vocales tiene.

1. **Elegir un día de la semana**

Crea una aplicación de escritorio donde:

* Muestres en una lista los días de la semana, con el formato (“Día 1: Lunes; Día 2: Martes; etc”)
* Recoge la opción elegida y muestra qué día escogió el usuario

1. **Suma de los números**

Crea una aplicación de escritorio donde ingreses un número entero y muestres la suma de los números que lo anteceden.

1. **Problema: Métodos por pantalla I**

Crea una aplicación de escritorio con cuadros de texto, etiquetas y cuadros de mensaje.

Declara e implementa los siguientes métodos:

1. **Método MostrarNombre** con un parámetro string. Al ejecutar el programa el nombre suministrado debe desplegarse en un cuadro de mensaje.
2. **Método MostrarNombres** con dos parámetros string que representen su

primer nombre y su apellido paterno. El método debe mostrar su primer nombre en un cuadro de mensaje, y su apellido paterno en otro.

1. **Método MostrarIngresos** con dos parámetros enteros que representen el salario de un empleado y el número de años trabajados. El método debe mostrar el total de ingresos obtenidos por el empleado en un cuadro de mensaje, suponiendo que haya obtenido la misma cantidad de ingresos cada año.
2. **Método CalcularPulgadas** que devuelva el equivalente en pulgadas de su argumento en centímetros.

La siguiente es una llamada de ejemplo:

**double pulgadas = CalcularPulgadas(10.5);**

(Pista: Multiplica los centímetros por 0.394 para calcular las pulgadas)

1. **Método CalcularAreaCirculo** que devuelva el área de un círculo, dado su radio como un argumento.

La siguiente es una llamada de ejemplo:

**double a = CalcularAreaCirculo(1.25);**

(Pista: El área del círculo se obtiene con base en la fórmula Math.PI \* r \* r. Aunque podríamos utilizar el número 3.14, Math.PI nos proporciona un valor más exacto.)

1. **Método EnSegundos** que acepte tres enteros, los cuales representarán el tiempo en horas, minutos y segundos. El método debe devolver el tiempo total en segundos.

La siguiente es una llamada de ejemplo:

**int totalSegundos = EnSegundos(1, 1, 2); // devuelve 3662**

1. **Método CalcularIncremento**, que sume 1 a su argumento entero.

La siguiente es una llamada de ejemplo:

**int n = 3;**

**int a = Incremento(n); // devuelve 4**

1. **Método CalcularSumaYDiferencia**, que calcule la suma y la diferencia de dos valores enteros cualesquiera (por ejemplo, si los argumentos de entrada son 3 y n, debe devolver 3+n y 3-n).
2. **Método CalcularSegundosAHMS**, que reciba un número de segundos y los convierta en horas, minutos y segundos.

Utiliza los operadores % y / (por ejemplo, 3662 segundos equivalen a 1 hora, 1 minuto y 2 segundos).

1. **Repartiendo cartas**

Crea un programa con un solo botón de manera que, cuando se haga clic en él, seleccione al azar una carta. Debes seguir el proceso:

* Primero, utiliza el generador de números aleatorios para crear un número en el rango de 1 a 4.
* Después, convierte el número en un palo (corazón, diamante, trébol y pica).
* Utiliza de nuevo el generador de números aleatorios para crear un número aleatorio en el rango de 1 a 13.
* Convierte el número en un as, un 2, 3, 5, etc.,
* Por último, muestra el valor de la carta elegida

1. **Ordenamiento**

Escribe un programa que reciba números de tres barras de seguimiento o tres

cuadros de texto y los muestre en orden, de menor a mayor.

1. **Entradas al cine**

Escribe un programa para averiguar cuánto paga una persona por asistir a una función de cine. El programa debe recibir una edad a partir de una TrackBar o caja de texto, y después tomar una decisión con base en estas condiciones:

* Si el espectador es menor de 5 años, la función es gratis;
* Si tiene de 5 a 12 años, paga la mitad de la tarifa;
* Si tiene de 13 a 64 años, paga la tarifa completa;
* Si es mayor de 65, la función es gratis.

1. **Apuestas**

Un grupo de personas desean apostar sobre el resultado de tres lanzamientos de dados. Cada una de ellas debe apostar 1€ para tratar de adivinar el resultado de los tres lanzamientos. Escribe un programa que utilice el método de números aleatorios para simular tres lanzamientos de un dado, y que muestre las ganancias de acuerdo con las siguientes reglas:

* si los tres dados caen en seis: la persona gana 20€;
* si los tres dados caen en el mismo número (pero no en seis): gana 10€;
* si dos de tres dados cayeron en el mismo número: gana 5€.

1. **Cerradura de combinación digital**

Escribe un programa que actúe como un candado de combinación digital para una caja fuerte:

* Crea tres botones que representen los números 1, 2 y 3. El usuario tiene que hacer clic en los botones para tratar de adivinar los números correctos (por ejemplo, 331121).
* El programa debe permanecer inactivo hasta que se opriman los botones correctos. Cuando esto suceda mostrará un mensaje de felicitación al usuario.
* Debe haber un botón para reiniciar
* Mejora el programa de manera que tenga otro botón para que el usuario pueda modificar la combinación de la caja fuerte.

1. **Juego de piedra, papel o tijera**

En este juego, dos jugadores eligen al mismo tiempo piedra, papel o tijera:

* La piedra gana a la tijera, el papel gana a la piedra, y la tijera gana al papel.
* Si ambos jugadores elijen la misma opción, es un empate.

Escribe un programa para representar este juego:

* El jugador debe seleccionar uno de tres botones marcados como piedra, papel o tijera.
* El ordenador tendrá que realizar su elección al azar mediante el uso del generador de números aleatorios; también debe decidir y mostrar quién gana.

1. **Despliegue de números enteros**

Escribe un programa que utilice un ciclo para mostrar los números enteros del 1 al 10, junto con sus cubos (es decir, el resultado de elevar cada uno a la tercera potencia).

1. **Números aleatorios**

Escribe un programa para mostrar diez números aleatorios utilizando un ciclo. Utiliza la clase Random para obtener números aleatorios en el rango de 1 a 100. Haz que los números se desplieguen en un cuadro de texto.

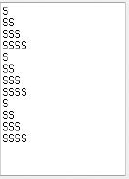
1. **Suma de enteros**

Escribe un programa que sume los números del 0 al 39 utilizando un ciclo. Comprueba que las respuestas obtenidas sean correctas; emplea para ello la fórmula para la suma de los números de 0 a n:

Suma = 5 x (n + 1) / 2

1. **Patrón de diente de sierra**

Escribe un programa para mostrar un patrón de diente de sierra en un cuadro de texto, como se muestra en la figura. El programa deberá utilizar la cadena "\r\n" para obtener una nueva línea.



1. **Tabla de multiplicar**

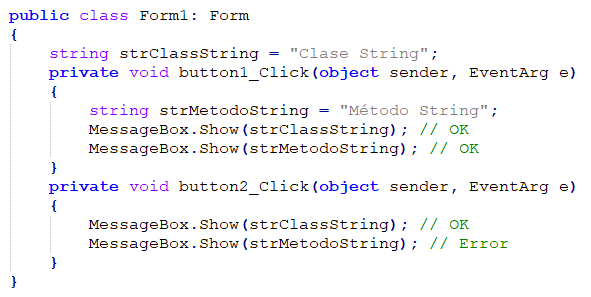
Escribe un programa para mostrar una tabla de multiplicación, como la que utilizan los niños pequeños. Además de utilizar la cadena "\r\n" para obtener una nueva línea, el programa debe usar la cadena "\t" para avanzar el tabulador a la siguiente posición, de manera que la información se muestre en columnas ordenadas.

Además, el programa debe ser capaz de mostrar una tabla de cualquier tamaño, el cual se debe especificar introduciendo un entero en un cuadro de texto.



1. **Visibilidad**

Coloca dos botones sobre un formulario y cambia el código en el button\_Click como se indica a continuación. Observa las regiones de código donde existen las variables tipo string.



Localiza las regiones de código donde puedes declarar string como public o private.

Declara una segunda ‘strClassString’ en la primera línea de ‘button2\_Click’, y observa qué ocurre:

