

CFGS

DESARROLLO DE APLICACIONES MULTIPLATAFORMA Y DESARROLLO DE APLICACIONES WEB

M03A: Programación.

Unidad Formativa 2: Programación modular.

UF2_PAC01_ENUNCIADO



UF2. [PAC 01]. Programación modular.

INFORMACIÓN

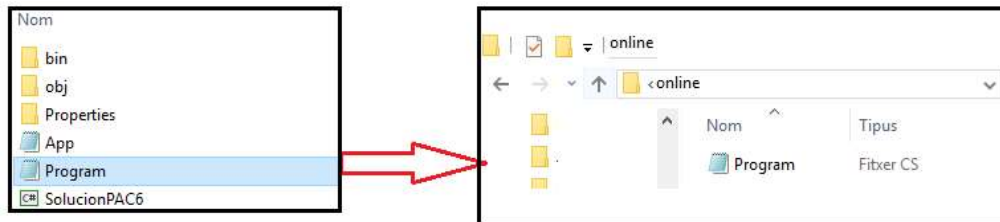
Para responder a las siguientes cuestiones deberás ayudarte del material didáctico y de internet.

Requisitos varios que deben cumplirse en vuestros trabajos:

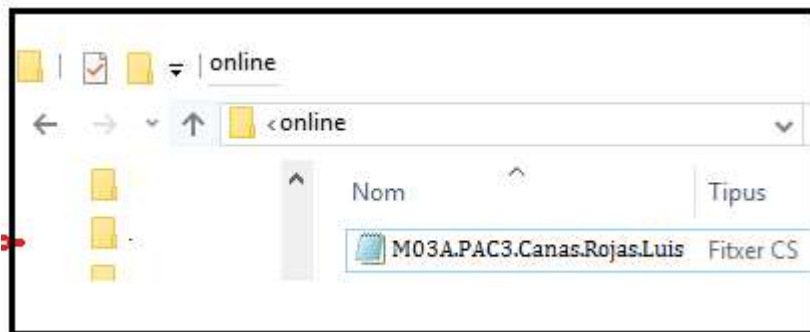
- En los ejercicios, si se requieren de cálculos, estos deben aparecer en la respuesta que planteéis.
- Siempre que utilicéis información de Internet para responder / resolver alguna pregunta, tenéis que citar la fuente (la página web) de dónde habéis sacado aquella información.
- Siempre que utilicéis información del material didáctico para responder / resolver alguna pregunta, tenéis que citar el tema y la página de dónde habéis sacado aquella información.
- No se aceptarán respuestas sacadas de Internet utilizando la metodología de copiar y pegar. Podéis utilizar Internet para localizar información, pero el redactado de las respuestas ha de ser vuestro.
- Las respuestas a las preguntas deben estar bien argumentadas, no se admiten respuestas escuetas o monosílabas.
- Se valorará la presentación de vuestro trabajo hasta con un punto y medio **(1)** de la nota final.

Instrucciones UF2 PAC1

- La fecha de entrega es improporrogable y la tenéis especificada en el campus.
- **La PAC se debe realizar en C#**
- En la primera línea del fichero, debe aparecer vuestro nombre, como comentario.
- Para hacer la entrega, cuando tengáis la PAC terminada, debéis hacer lo siguiente:
 - Copiar el fichero Program.cs de vuestro proyecto, a donde queráis, por ejemplo a una carpeta que se llame "online":



- Cambiar el nombre del fichero Program.cs, al siguiente: **M03A.UF2.PAC1.Apellido1.Apellido2.Nombre**. Los apellidos y nombres se escriben sin acentos, espacios, eñes....Por ejemplo, el alumno Luis Cañas Rojas, debe usar el siguiente nombre para el fichero de la PAC1: M03A.UF2.PAC1.Canas.Rojas.Luis.



- Éste es el fichero que debéis subir al campus.

Evaluación genérica de los ejercicios

Los ejercicios entregados deben **compilar** para ser evaluados. Para cada ejercicio se verificará todo lo que se expone a continuación.

- **Trabajo con funciones (40%):** Debéis realizar las funciones pedidas, tal y como se os indica en el enunciado. **Esto es obligatorio. Si no trabajáis con funciones, no se puntúa la PAC**
- **Funcionalidad(35%):** El programa debe mostrar resultados correctos para entradas razonables. Haced pruebas en casos "críticos" o "fronterizos". Por ejemplo, si el programa debe pedir la edad del usuario, obviamente introduciré un número cuando yo lo pruebe(no letras), pero... qué pasa si entro el -7? tiene sentido?
- **Comentarios (10%):** El código debe ser comentado, explicando la manera que habéis usado para programar vuestra solución al problema.
- **Elegancia y eficiencia de la solución(15%):** el programa no debe hacer operaciones que malgasten tiempo de cálculo. El código debe estar bien estructurado, con nombres de variables claros.

Prohibiciones

No se pueden usar ni arrays ni ficheros

Enunciado UF2 PAC1

1.-Realiza un programa que muestre un menú con opciones del 1 al 4. Cada opción debe pedir los parámetros que se enviarán a la función (desde el MAIN), y llamar a la función correspondiente, y mostrar los resultados en los casos que se pide. El programa debe ir ejecutándose hasta que no escojamos la opción 4

- Opcion1: La función será del tipo void (no devuelve nada), y recibe tres parámetros enteros. En el main se deben pedir tres números por teclado, y mandarlos a la función. Dentro de esta función se deben imprimir por pantalla todos los números entre el primero y el segundo (de mayor a menor), que sean múltiplos del tercero.
- Opcion2: Desde el main pediremos dos cadenas de caracteres (string). La función recibirá las dos cadenas como parámetros de entrada, una cadena como parámetro de salida, y devolverá un entero. La función deberá convertir primeramente las dos cadenas a mayúsculas. Luego, debe generar una nueva cadena (la que se devuelve por parámetro de salida) que debe cumplir lo siguiente:
 - Debe tener una longitud superior o igual a 5 veces la longitud de la primera cadena.
 - La nueva cadena estará formada por la cadena1 seguida de un espacio, seguida de la cadena2, seguida de un espacio, tantas veces como sea necesario hasta que se cumpla el punto anterior.
 - La función devolverá el número de veces que se ha realizado la operación anterior.
- Opcion3: Desde el main generaremos dos números aleatorios entre 1 y 100 incluidos, y los mandaremos a la función, la cual recibe estos 2 números por valor, y otros 3 parámetros por referencia. La función pondrá un 0 o un 1 en el tercer, cuarto y quinto parámetro, en función de si la suma de los dos primeros números es múltiplo de 2, de 3 y de 5 respectivamente (un 0 si no es múltiplo y un 1 si es múltiplo). Luego, desde el main, se debe imprimir un mensaje indicando si es múltiplo de 2, 3 y/o 5, dependiendo del valor de los 3 parámetros.
- Opcion4: esto finalizará el programa. No hace falta implementar ninguna función.

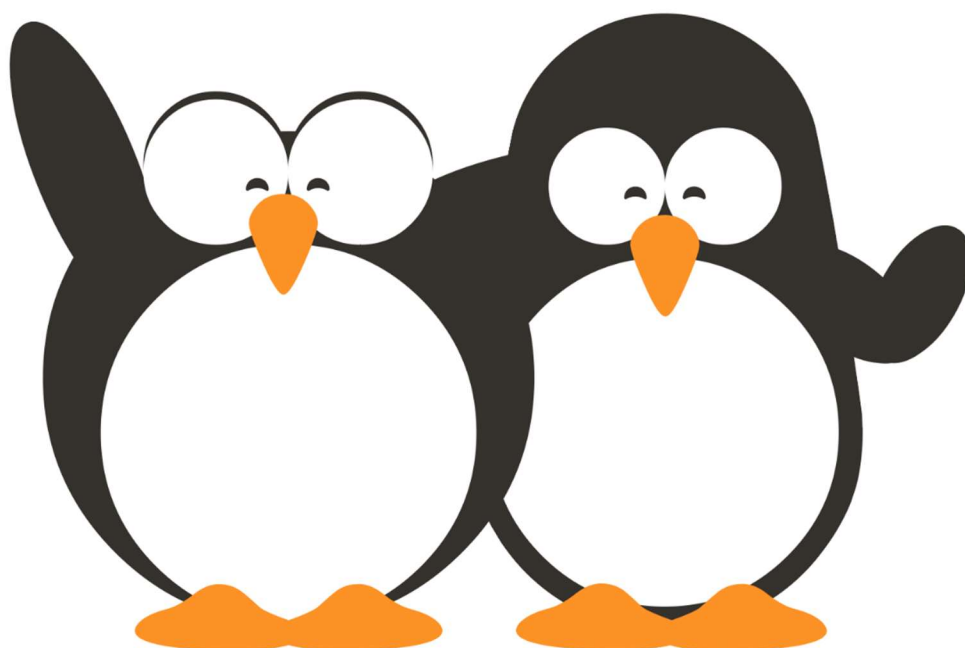
Ejemplos de ejecución de cada opción (en negrita lo q escribe el usuario):

```
//opción 1, pido desde el main
Introduce numero1:
34
Introduce numero2:
17
Introduce numero3:
4
//ahora llamo a la función, y la función imprime los múltip
los de 4 entre 34 y 17
32
28
24
20

//opción 2, todo se lee y se imprime desde el main
Introduce texto:
Adios
Introduce texto:
Coche
ADIOS COCHE ADIOS COCHE ADIOS COCHE
3 REPETICIONES

//opción 3, supongamos que los aleatorios son el 3 y el 7 /
/Desde el main imprimiremos
Aleatorios: 3 y 7
La suma es múltiple de 2
La suma NO es múltiple de 3
La suma es múltiple de 5
```

¡Buen trabajo!



www.ilerna.es