



NOTAS PARA LA REALIZACIÓN DEL EXAMEN:

- El **alumnado** que **NO** haya seguido la **Evaluación Continua**, debe ponerse en contacto con su profesor/a para realizar el **test** correspondiente.
- La solución se almacenará en la carpeta **FP\EXAMENJUL**, dentro de Documentos. Si la carpeta ya existe, debe borrarse todo su contenido. En otro caso, debe crearse.
- Los nombres de los ficheros con la solución para los ejercicios 1, 2 y 3 serán **ejercicio1.cpp**, **ejercicio2.cpp** y **ejercicio3.cpp**, respectivamente.
- Al inicio del contenido de cada fichero deberá aparecer un comentario con **el nombre del alumno o de la alumna, titulación, grupo y código del equipo** que se está utilizando (cada dato en una línea diferente).
- **La primera instrucción de la función main() de cada ejercicio será un cout** para mostrar por pantalla la información a la que se hace referencia en el punto anterior.
- **Debe consultarse** el documento **“Obligaciones y Recomendaciones Estilo de Programación”**, disponible en la Sala Común del Campus Virtual de la asignatura, con objeto de tener en cuenta los puntos allí señalados en las soluciones a los ejercicios.
- **Hay que asegurarse** de que las **Opciones de Compilación** que se recomiendan en la última página del documento **“Guía de Uso de Code::Blocks”** (-Werror, _GLIBCXX_DEBUG, -Wall, -Wextra, -O2), disponible en la Sala Común del Campus Virtual de la asignatura, **están activadas** en el entorno Code::Blocks. Si no es así, deberían activarse.
- Una vez terminado el examen, se subirán los ficheros ***.cpp** a la tarea creada en el **campus virtual** para ello.
- **No está permitido:**
 - Utilizar documentación electrónica o impresa.
 - Intercambiar documentación con otros compañeros.
 - Utilizar soportes de almacenamiento.
 - Utilizar dispositivos electrónicos (móviles, tablets, ...)

(2.5 ptos) 1.- **Diseña una función** `mayorLongitud` que recibe como parámetro un array unidimensional de tamaño `TAM` (una constante definida) relleno de números enteros y devuelve la longitud (número de valores) de la mayor sub-sucesión de números ordenados. A continuación, **diseña un programa principal** (función `main`) para probar el funcionamiento de dicha función. El programa leerá por teclado una secuencia de `TAM` números enteros y mostrará por pantalla la longitud de la mayor sub-sucesión ordenada de dicha secuencia.

Ejemplo 1: Con los valores del array de entrada (para `TAM = 10`): 42 7 56 87 64 2 35 8 9 0, obtendríamos las siguientes sub-sucesiones ordenadas: 42 ; 7 56 87 ; 64 ; 2 35 ; 8 9 ; 0, siendo la mayor sub-sucesión ordenada: 7 56 87, de longitud 3.

Ejemplo 2: Con los valores del array de entrada (para `TAM = 10`): 12 21 23 1 23 3 2 43 43 333, obtendríamos las siguientes sub-sucesiones ordenadas: 12 21 23 ; 1 23 ; 3 ; 2 43 43 333, siendo la mayor sub-sucesión ordenada: 2 43 43 333, de longitud 4.

Para estos ejemplos la ejecución del programa mostraría por pantalla lo siguiente:

```
Introduzca 10 numeros enteros: 42 7 56 87 64 2 35 8 9 0
La longitud de la mayor sub-sucesion es: 3
```

```
Introduzca 10 numeros enteros: 12 21 23 1 23 3 2 43 43 333
La longitud de la mayor sub-sucesion es: 4
```

(3.5 ptos) 2.- Diseña un algoritmo que lea de teclado primero una cadena de caracteres que constituirá un determinado patrón y después un texto. El algoritmo mostrará por pantalla un listado de todas las palabras del texto cuya suma de valores de sus caracteres en ASCII coincida con la suma de los valores de los caracteres de la cadena patrón. **En la salida no habrá palabras repetidas.**

Ejemplo:

Entrada:

```
Introduzca el patrón: CASA
Introduzca el texto (FIN para terminar):

SACA DE CASA EL CAKI SOLO BEBO AGUA EN CASA CON EL CAKI
FIN
```

Salida:

```
Las palabras que cumplen la condición son:

SACA CASA CAKI BEBO
```

NOTAS:

- El texto contiene un número indefinido de palabras.
- El texto termina con la palabra FIN.
- Cada palabra tiene un número indefinido pero limitado de caracteres (todos alfabéticos mayúsculas).
- Las palabras del texto estarán separadas, indistintamente, por espacios en blanco y/o por saltos de línea.
- En el texto aparecerá un número máximo de 15 palabras distintas (define una constante MAX_PAL_DIST con ese valor).

(4 ptos) 3.- Una imagen captada por una cámara digital puede representarse como una matriz a de tamaño $N \times M$, donde N es la resolución vertical, M la horizontal y el valor $a[i][j]$ es un número natural proporcional a la intensidad luminosa en el punto de coordenadas (i, j) . Una operación típica que se hace con este tipo de imágenes es el suavizado para eliminar ruidos. Para ello, el valor en cada punto se sustituye por la media entera (sin decimales) del valor almacenado en ese punto y el de sus vecinos. Hay que tener en cuenta que un punto puede tener diferente número de vecinos dependiendo del lugar que ocupe en la imagen (matriz). Así, los puntos que están en las esquinas de la matriz tienen solo 3 vecinos, los puntos que están en un borde de la matriz sin ser esquina tienen 5 vecinos y el resto de puntos tienen 8 vecinos.

- **Define** el **tipo** de dato `TImagen` para una matriz de números naturales de tamaño $N \times M$ (N y M dos constantes)
- **Diseña** un **procedimiento** suavizado que reciba como parámetro de entrada una matriz `a` de tipo `TImagen` que almacena una imagen digital y devuelva como parámetro de salida una matriz `b` del mismo tipo, resultado de llevar a cabo la operación promedio antes descrita sobre cada uno de los valores de `a`. Más abajo puedes ver un ejemplo.
- **Diseña** un **programa principal** (función `main`) para probar el funcionamiento de dicho procedimiento. El programa leerá por teclado una matriz de $N \times M$ números naturales y mostrará por pantalla la matriz resultado de realizar el suavizado de la misma.

Ejemplo:

Para la matriz de entrada:

3	2	5
7	8	6
9	3	2
4	3	1

Se obtendrá la matriz de salida:

5	5	5
5	5	4
5	4	3
4	3	2

Ejemplos de cálculo:

- Para la celda (2,1) de la matriz `a` con valor 3, el cálculo del promedio sería la suma de ese mismo valor más los valores de sus 8 vecinos, dividida entre 9: $(7+8+6+9+3+2+4+3+1)/9=4$ (división entera). Por ese motivo, el valor de la celda (2,1) en la matriz `b` es 4.
- Para la celda (2,2) de la matriz `a` con valor 2, el cálculo del promedio sería la suma de ese mismo valor más los valores de sus 5 vecinos, dividida entre 6: $(8+6+3+2+3+1)/6=3$ (división entera). Por ese motivo, el valor de la celda (2,2) en la matriz `b` es 3.

Nota:

- Se recomienda no utilizar el tipo `unsigned`, sino el tipo `int` para trabajar con valores de tipo entero y natural.