



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\EXAMENFEB**, 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 pts) 1.- El triángulo de Pascal es un triángulo de números enteros mayores o iguales que 1, infinito y simétrico. Se empieza con un 1 en la primera fila, y en las filas siguientes los números de los extremos son un 1 y cada uno de los restantes números se calcula como la suma de los dos números que tiene encima. Así, el triángulo con 7 filas sería:

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
 1 5 10 10 5 1
1 6 15 20 15 6 1
```

El triángulo de Pascal se puede almacenar en una matriz cuadrada girado 45° en el sentido contrario a las agujas del reloj, de forma que cada elemento de la primera fila y la primera columna de la matriz es un 1, los elementos a la derecha de la diagonal

secundaria o inversa son un 0 y cada uno de los restantes números se calcula como la suma de los dos números que tiene a su izquierda y encima. Así, el triángulo anterior quedaría almacenado de la siguiente forma en una matriz 7x7:

1	1	1	1	1	1	1
1	2	3	4	5	6	0
1	3	6	10	15	0	0
1	4	10	20	0	0	0
1	5	15	0	0	0	0
1	6	0	0	0	0	0
1	0	0	0	0	0	0

Diseña un algoritmo que, para una constante dada TAM, construya el triángulo de Pascal en una matriz cuadrada TAMxTAM y muestre por pantalla el contenido de dicha matriz (los valores 0 no se muestran). Así, la ejecución del algoritmo para TAM = 7, mostraría por pantalla los siguientes números:

1	1	1	1	1	1	1
1	2	3	4	5	6	
1	3	6	10	15		
1	4	10	20			
1	5	15				
1	6					
1						

(3.5 pts) 2.- Diseña un algoritmo que lea de teclado un texto, y muestre por pantalla un listado de todas las palabras del texto cuyos caracteres estén ordenados alfabéticamente en orden creciente. Puede haber más de una ocurrencia del mismo carácter en una palabra. **En la salida no habrá palabras repetidas.**

Ejemplo:

Entrada:

Introduzca el texto (FIN para terminar):

ELLOS CANTAN EL HIMNO DE SU EQUIPO DE FUTBOL CON PASION
Y YO NO CANTO EL HIMNO FIN

Salida:

Las palabras cuyos caracteres estan ordenados son:
ELLOS EL HIMNO DE SU Y NO

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.- Deseamos simular el proceso de “Escalado de Imágenes” que muchos dispositivos (ej. televisiones) llevan a cabo para aumentar o disminuir la resolución de las mismas. Pensemos la imagen como una matriz M de F filas y C columnas y cuyo contenido en cada celda es un número entero mayor o igual que 1 (representando un determinado color). Para aumentar la resolución de la imagen, se construye una nueva matriz T con el doble de filas de M y con el mismo número de columnas. Para rellenar las $2 \cdot F$ filas de la matriz T se realiza el siguiente proceso:

- Primero se calcula la media entera (sin decimales) de todos los valores de M . Llamemos a este valor $mediaM$.
- Las filas de M pasan a ser las filas pares de T (la fila 0 de M se copia en la fila 0 de T , la fila 1 de M se copia en la fila 2 de T y así sucesivamente).
- Las filas impares de T se inicializan a 0. Posteriormente, se calculan los valores a almacenar en cada celda de cada fila impar de T de la siguiente manera: se recorre la fila de izquierda a derecha y en cada celda se almacena la media entera de los valores de los vecinos de esa celda (si alguno de esos vecinos es un 0, se toma como valor de ese vecino la media $mediaM$ antes calculada. Ver tercer paso del ejemplo mostrado a continuación).


Nota: Las celdas de las esquinas tienen sólo 3 vecinos; las de los bordes que no son esquinas tienen 5 y las restantes tienen 8. Hay que controlar bien esos cálculos.

Veamos un ejemplo:

Matriz M

Matriz T

3	2	5
6	5	3



3	2	5
4	4	3
6	5	3
5	4	4

Pasos para conseguir T

- Calculamos $mediaM = 4$ $((3+2+5+6+5+3) / 6 = 4)$
- Tras copiar las filas de M a las filas pares de T e inicializar las impares a 0 la matriz T queda de la siguiente forma:

3	2	5
0	0	0
6	5	3
0	0	0

- La primera fila impar de T se calcula de la siguiente forma:
 - La celda (1,0) se calcula a partir de sus vecinos que son 5 (3, 2, 0, 6, 5). Tras sustituir el 0 por 4 (que es $mediaM$), sumarlos y dividir el resultado entre 5 nos da el valor 4 $((3+2+4+6+5) / 5 = 4)$. En la celda (1,0) ponemos por tanto un 4.
 - La celda (1,1) tiene 8 vecinos de los cuales uno es 0, que se sustituye por 4; la suma de todos es 32; que dividido entre 8 vecinos resulta $32/8=4$. En la celda (1,1) ponemos por tanto un 4.
 - Para la celda (1,2) se suman sus 5 vecinos y el valor resultante es 3 $((2+5+4+5+3) / 5 = 3)$.

Tras estos cálculos la matriz T queda:

3	2	5
4	4	3
6	5	3
0	0	0

- Realizando un proceso similar para la otra fila impar, terminamos la matriz T:

3	2	5
4	4	3
6	5	3
5	4	4

Diseña un algoritmo que, para dos constantes dadas F y C , lea de teclado el contenido de la matriz M (F filas y C columnas), construya la matriz T ($2 * F$ filas y C columnas) y muestre por pantalla el contenido de la matriz T . Así, la ejecución del algoritmo para $F = 2$, $C = 3$ y la matriz M ejemplo que hemos usado antes, mostraría por pantalla:

```
Introduzca la matriz M (2x3):  
3 2 5  
6 5 3  
  
La matriz T (4x3) es la siguiente:  
3 2 5  
4 4 3  
6 5 3  
5 4 4
```

Notas:

- Se recomienda no utilizar el tipo `unsigned`, sino el tipo `int` para trabajar con valores de tipo entero y natural.
- Se recomienda ir probando la ejecución del algoritmo conforme se van acometiendo los diferentes pasos de la construcción de la matriz T .