



UNIVERSIDAD DE MÁLAGA
Dpto. Lenguajes y Ciencias de la Computación
E.T.S.I. Informática

Fundamentos de la Programación
Examen 2ª Convocatoria Ordinaria

01/09/17

Apellidos, Nombre:

Titulación:

Grupo:

Código PC usado:

NOTAS PARA LA REALIZACIÓN DEL EXAMEN:

- La solución se almacenará en el directorio **C:\FP\EXAMENSEPT**. Si el directorio 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, 3 y 4 serán **ejercicio1.cpp**, **ejercicio2.cpp**, **ejercicio3.cpp** y **ejercicio4.cpp**, respectivamente.
- Al inicio del contenido de cada fichero deberá aparecer un comentario con **el nombre del alumno, titulación, grupo y código del equipo** que se está utilizando (cada dato en una línea diferente).
- 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, ...)

(1 pto) 1.- Diseña una función `sumaMayoresImpares` que recibe como parámetro una matriz de `FIL` filas y `COL` columnas (`FIL` y `COL` dos constantes definidas) rellena de números naturales y devuelve la suma de los mayores números impares de cada fila de la misma. Si en una fila no hay números impares, el valor de esa fila para la suma es 0. Si en una fila el mayor número impar se repite varias veces, sólo se tiene en cuenta su valor una vez para la suma.

Importante: No debes modificar el código ya proporcionado en el campus virtual (*ejercicio1.cpp*) para probar la función. Debes completar el código de la función `sumaMayoresImpares`. Si para su diseño necesitas más procedimientos o funciones, puedes añadirlos. Si el código no funciona correctamente la puntuación de este problema será de 0 puntos.

La ejecución del código suministrado una vez diseñada la función `sumaMayoresImpares` mostrará por pantalla lo siguiente:

```
El contenido de la matriz de prueba es:
```

```
6 4 12 2
5 2 7 3
4 9 5 11
```

```
La suma de los mayores impares de las filas es: 18
```

(2 ptos) 2.- Diseña una función `mayorLongitud` que recibe como parámetro un array unidimensional de tamaño `TAM` (una constante definida) relleno de números naturales 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 naturales y mostrará por pantalla la longitud de la mayor sub-secuencia 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-secuencias ordenadas: 42 ; 7 56 87 ; 64 ; 2 35 ; 8 9 ; 0, siendo la mayor sub-secuencia 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-secuencias ordenadas: 12 21 23 ; 1 23 ; 3 ; 2 43 43 333, siendo la mayor sub-secuencia ordenada: 2 43 43 333, de longitud 4.

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

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

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

(3.5 ptos) 3.- 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).
- En el texto aparecerán un número máximo MAX_PAL_DIST (una constante) de palabras distintas.
- El carácter separador de palabras es el espacio en blanco.

(3.5 ptos) 4.- 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.