



Práctica final

(1ª Convocatoria)

Entrega 2/3

Ficheros de Texto



- Tratamiento del fichero “.csv” de entrada

En **UBUVirtual** tenéis un fichero de nombre `DatosTelefonosMoviles.csv` el cual deberíais abrir con OpenOffice, con “opciones de separador” de “comas”, y que está constituido por una tabla formada por 1000 filas y 15 columnas, tal como:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	battery_power	blue	clock_speed	dual_sim	fc	four_g	int_memory	m_dep	mobile_wt	n_cores	pc	ram	sc_h	sc_w	brand
2	1043	11.8		1	14	0	50.1		193	3	16	3476	12	7	Nokia
3	841	10.5		1	4	1	610.8		191	5	12	3895	6	0	Xiaomi
4	1807	12.8		0	1	0	270.9		186	3	4	2396	17	10	Samsung
5	1546	00.5		1	18	1	250.5		96	8	20	3893	10	0	LG
6	1434	01.4		0	11	1	490.5		108	6	18	1773	15	8	ZTE
7	1464	12.9		1	5	1	500.8		198	8	9	3506	10	7	Nokia
8	1718	02.4		0	1	0	47	1	156	2	3	3873	14	2	ZTE
9	833	02.4		1	0	0	620.8		111	1	2	1495	7	2	ZTE
10	1111	12.9		1	9	1	250.6		101	5	19	3485	11	9	Nokia
11	1520	00.5		0	1	0	250.5		171	3	20	651	6	0	Samsung
12	1500	02.2		0	2	0	550.6		80	7	6	3866	13	7	HTC
13	1343	02.9		0	2	1	340.8		171	3	6	3911	15	8	ZTE
14	900	11.4		1	0	0	30	1	87	2	3	439	6	2	Samsung
15	1190	12.2		1	5	0	190.9		158	5	15	992	13	0	HTC
16	630	01.8		0	8	1	510.9		193	8	9	2751	17	6	Xiaomi
17	1846	1	1	0	5	1	530.7		106	8	7	563	9	5	Nokia
18	1985	00.5		1	14	1	26	1	163	2	17	2083	13	3	Apple
19	1042	02.9		0	5	1	480.2		186	4	15	2187	9	2	Nokia
20	1231	11.7		1	2	1	370.2		194	2	3	3902	19	12	Samsung
21	1488	02.6		0	9	0	370.7		189	4	20	2524	5	0	Xiaomi
22	968	00.6		0	8	1	70.7		151	1	17	1357	15	1	ZTE
23	529	02.6		1	1	0	600.5		101	5	5	3456	13	11	Xiaomi
24	1558	01.7		1	7	0	500.1		115	2	10	1641	17	0	LG
25	533	10.7		1	16	0	580.8		97	5	18	2322	17	3	LG
26	1037	01.7		1	1	0	50.7		125	3	6	3862	17	4	Xiaomi



- La tabla contiene datos de dispositivos móviles, en concreto cada columna contiene:
 - **battery_power**: Potencia de la batería (mAh).
 - **blue**: Si dispone o no de bluetooth.
 - **clock_speed**: Velocidad del reloj del procesador.
 - **dual_sim**: Si dispone o no de SIM dual.
 - **fc**: Megapíxeles de la cámara frontal.
 - **four_g**: Si dispone o no de 4G.
 - **int_memory**: Memoria interna (GB).
 - **m_dep**: Grosor del móvil (cm).
 - **mobile_wt**: Peso del móvil.
 - **n_cores**: Número de cores.
 - **pc**: Megapíxeles de la cámara principal.
 - **ram**: Memoria RAM (MB).
 - **sc_h**: Altura de la pantalla en megapíxeles.
 - **sc_w**: Anchura de la pantalla en megapíxeles.
 - **brand**: Fabricante del móvil.



- Para poder tratar dichos datos, consideraremos:
 - **battery_power**: Entero; Valor en el rango [0; 5000].
 - **blue**: Booleano.
 - **clock_speed**: Decimal; Valor en el rango [0; 4.0].
 - **dual_sim**: Booleano.
 - **fc**: Entero; Valor en el rango [0; 20].
 - **four_g**: Booleano.
 - **int_memory**: Entero; Valor en el rango [2; 128].
 - **m_dep**: Decimal; Valor en el rango [0.1; 2.0].
 - **mobile_wt**: Entero; Valor en el rango [50; 500].
 - **n_cores**: Entero; Valor en el rango [1; 16].
 - **pc**: Entero; Valor en el rango [0; 40].
 - **ram**: Entero; Valor en el rango [128; 2560].
 - **sc_h**: Entero; Valor en el rango [0; 100].
 - **sc_w**: Entero; Valor en el rango [0; 100].
 - **brand**: Cadena de Caracteres.



- Tratamiento del fichero “.csv” de entrada

Para esta entrega, se pide escribir un programa en C con las siguientes opciones, que deben aparecer en forma de menú:

```
*****
```

1. Número total de caracteres alfanuméricos (es decir, solo números y letras)
2. Número total de filas, sin contar la cabecera.
3. Fila más larga (es decir, con el mayor número de caracteres).
4. Añadir nueva fila (al final del archivo).
0. Salir

```
*****
```



- **Para añadir una fila al fichero:**
 - El programa debe preguntar secuencialmente cada dato mostrando el rango admitido para cada variable.
 - Se debe comprobar que cada variable introducida sea válida y en caso contrario volver a pedir ese dato.
 - Ten en cuenta que hay datos que son cadenas de caracteres (Columna 15)
 - Una vez solicitados y validados todos los datos se genera un nuevo archivo actualizado con todos los datos existentes más la nueva fila que se llamará:

`DatosTelefonosMoviles_v2.csv`



- **Prototipos y funciones mínimos:**

- */*Muestra el menú y solicita una de las opciones hasta que introduzcamos una opción válida*/*
`int solicitarOpcionMenu();`
- */*Llama a las funciones que ejecutan cada una de las opciones del menú mediante un switch. Abrir y cerrar el fichero desde esta función y pasarlo ya abierto al resto de funciones que lo utilizan*/*
`void seleccionarOpcion(int);`
- */*Cuenta los caracteres del fichero que le llega por parámetro.*/*
`int contarCaracteres(FILE*);`
- */*Cuenta el número total de filas del fichero*/*
`int numeroTotalFilas(FILE*);`
- */*Devuelve el índice de la fila con más caracteres*/*
`int filaMaslarga(FILE*);`



- **Prototipos y funciones mínimos:**

- */*La llamaremos cada vez que haya que validar la entrada de un entero por teclado.*

Datos de entrada (parámetros) de la función:

- *@param num : valor a validar*
 - *@param lim_inf : límite inferior del rango admitido*
 - *@param lim_sup : límite superior del rango admitido*
 - *@param leidos : número de valores leídos por teclado que devuelve scanf (han de ser dos: uno para el dato introducido, y el otro el “enter”)*
 - *@param enter : salto de línea introducido por teclado*

Datos de salida (lo que devuelve la función):

- *@return resultado de la validación (Booleano)*

**/*

bool validarEntero(int, int, int, int, char);



- **Prototipos y funciones mínimos:**

- */*La llamaremos cada vez que haya que validar la entrada de un real por teclado.*

Datos de entrada (parámetros) de la función:

- *@param num : valor a validar*
- *@param lim_inf : límite inferior del rango admitido*
- *@param lim_sup: límite superior del rango admitido*
- *@param leidos : número de valores leídos por teclado que devuelve scanf
(han de ser dos: uno para el dato introducido, y el otro el “enter”)*
- *@param enter :salto de línea introducido por teclado*

Datos de salida (lo que devuelve la función):

- *@return resultado de la validación (Booleano)*

**/*

```
bool validaReal(float, float, float, int, char);
```



- **Prototipos y funciones muy recomendables:**

- */*Muestra un menú con los 7 nombres de los fabricantes (marcas) de móvil identificadas con enteros [1-7] y solicita una de las opciones hasta que introduzcamos una válida*/*

```
int menuMarcasMoviles();
```

1. Apple
2. HTC
3. LG
4. Nokia
5. Samsung
6. Xiaomi
7. ZTE



- **Requisitos básicos**

1. La práctica debe compilar para que sea corregida.
2. El código debe estar debidamente documentado, con comentarios internos: comentarios de cabecera del programa, comentarios de cabecera de las funciones, etc.
3. Validar la opción del menú: entrada robusta (véase Tema 7: Sentencias Repetitivas).
4. Leer el fichero completo, independientemente de las filas que tenga.
5. Utilizar siempre la lectura adelantada para el tratamiento de ficheros (véase Tema 8: Ficheros).
6. Abrir y cerrar el fichero en cada opción del menú; y pasarlo siempre a la función correspondiente (véase Tema 8: Ficheros).
7. Verificar siempre el posible contenido vacío del fichero o cualquier otro error.
8. Utilizar variables *booleanas* siempre que sea necesario.
9. No se permite utilizar variables globales.
10. Utilizar tantas constantes como sea necesario (nombre de ficheros; tamaño de cadenas; nombres de fabricantes...)

- **Consideraciones:**

Las dudas serán resueltas en el [Foro Dudas Práctica Obligatoria de UBUVirtual](#), en clase de prácticas, y en tutorías. Nunca por correo electrónico.