

**Objetivo de la práctica:**

Practicar uso de ficheros: abrir, cerrar y tratamiento de información contenida en el fichero

**Uso de Ficheros**

1. No olvidar incluir la cabecera: `#include <fstream.h>`
2. Los ficheros se pueden abrir para leer su contenido o para escribir en ellos. Nunca leeremos y escribiremos simultáneamente (nota: *fichero* es solamente el nombre de la variable):

Clase para definición de un objeto fichero de escritura: `ofstream fichero;`  
 Clase para definición de un objeto fichero de lectura: `ifstream fichero;`  
 Apertura del fichero: `fichero.open ("nombre_fichero");`  
 Cierre del fichero: `fichero.close ();`

Nota: por defecto el Dev C++ deja el fichero creado en el mismo directorio donde está el compilador:

```
fichero.open("c:\\tmp\\hola.txt"); //deja en fichero hola.txt en el directorio c:\tmp (notad la doble \)
```

3. Escritura y lectura. La lectura y escritura de información se realiza a través los operadores `<<` y `>>` (de forma equivalente al uso de `cin` y `cout`):

```
fichero << dato; //donde dato es cualquier tipo de dato convertido a carácter. Escritura en fichero
fichero >> dato; //lectura desde fichero
```

4. Fin de fichero. Para leer un fichero hasta su fin se empleará:

```
while (fichero >> dato) // esta condición proporcionará FALSE cuando se llegue al final.
```

5. Estructuras. No se puede leer ni almacenar una estructura directamente en un fichero, sino que debe procederse por separado con cada uno de sus campos:

```
ejemplo: struct complejo
{
    int real;
    int imag;
}
```

```
Bien: salidafich << complejo.real << " " << complejo.imag;
Mal: salidafich << complejo;
```

**PROBLEMAS**

- 1) Escribe un programa que cuente el número de veces que aparece en un archivo cada una de las letras del alfabeto, sin distinguir mayúsculas de minúsculas. Para el resto de caracteres (números, espacios, puntos, comas, etc) utilizaremos el mismo contador. La salida por pantalla tiene que ser del tipo

Carácter	numero de veces
A	10
B	5
.....	
Z	0
Otros	7



- 2) Uno de los primeros algoritmos de cifrado que se conoce es atribuido al emperador romano Julio César. El algoritmo consiste en sustituir cada una de las letras por la situada 3 posiciones más allá en el alfabeto, permaneciendo el resto de caracteres sin ninguna modificación. Así:

'a' se sustituye por la 'd',  
'b' por la 'e'  
...  
...  
'x' por la 'a',  
'y' por 'b',  
'z' por 'c'

Realizar un programa que lea un fichero de texto y escriba en otro fichero el texto codificado con el algoritmo de Julio César.

- 3) Escribe un programa que compare dos ficheros de texto línea a línea y escriba por pantalla la palabra iguales si ambos ficheros son iguales o el número de línea en la que se produce la primera diferencia entre los textos de los ficheros.
- 4) Un fichero binario de nombre `articulos.dat` contiene un número indeterminado de registros de artículos. Cada registro tiene asociada la siguiente estructura:

```
struct articulo
{
    int codigo,
    int existencias,
    float precio
}
```

Escribe una función para cada una de las siguientes tareas:

- Listar por pantalla los artículos cuyas existencias sean cero.
- Añadir un nuevo artículo al fichero, teniendo en cuenta que el código del artículo no debe existir previamente en el fichero
- Escribir el contenido del fichero en un nuevo fichero en formato HTML. El fichero de salida tendrá el nombre `articulo.html` y su estructura será :

```
<HTML>
<BODY>
<P>
<B>Codigo: </B> Codigo_articulo <BR>
<B>Existencias: </B> Existencias_articulo <BR>
<B>Precio: </B> Precio_articulo
</P>
...
...
</BODY>
</HTML>
```