

Modificaciones

(1.3)

Diseño de Referencia

Cambios:

- Cuando se hace una copia de un fichero en disco (lo que anteriormente hacía *copyFile*), se quiere **normalizar los saltos de línea** (quitar '\r' y dejar solo los '\n').
- Al mandar por Internet, se quiere **encriptar** (+1 si es letra o dígito).
- Al mandar por Bluetooth, se quiere **encriptar** (de igual forma que antes) y eliminar **espacios** repetidos.

Modificación 1

Cuando se hace una copia de un fichero en disco (lo que anteriormente hacía *copyFile*), además de *normalizar saltos de línea* (quitar `\r`), hay veces que también se querrá encriptar. En el *main* se debería poder indicar si, al hacer la copia, se quiere encriptar o no.

```
void main () {  
    ...  
    fs.copy("privado.txt" ... "a.txt");    // Poder indicar que "a.txt" quede sin encriptar  
    fs.copy("privado.txt" ... "b.txt");    // Poder indicar que "b.txt" quede encriptado  
}
```

Modificacion 2

Al enviar por Internet, además de encriptar, se quiere poder indicar si hay que normalizar (quitar '\r') y/o eliminar espacios repetidos (una de ellas o las dos).

```
void main () {  
    ...  
    fs.copy("privado.txt" ... "1.1.1.1");    // Este envío normalizado (sin '\r')  
    fs.copy("privado.txt" ... "1.1.1.2");    // Este envío sin espacios repetidos  
    fs.copy("privado.txt" ... "1.1.1.3");    // Este normalizado y sin espacios repetidos  
}
```

Modificacion 3

En los tres casos (tanto si se copia el fichero en disco como si se envía por Internet o Bluetooth) se quiere saber, desde el Main, el número de caracteres copiados.

```
void main () {  
    ...  
    fs.copy("privado.txt" ... "1.1.1.1");  
    System.out.println("Se han enviado " + ... + "# caracteres");  
}
```

Modificacion 4

Tanto al normalizar (quitar ‘\r’) como al eliminar espacios repetidos, se quiere saber también cuántos caracteres había antes del procesado.

```
void main () {  
    ...  
    fs.copy("privado.txt" ... "1.1.1.1");           // Se envía normalizado (sin ‘\r’)  
    System.out.println("Inicialmente había “ + ... + “# caracteres”);  
    System.out.println("Después de normalizar, se han enviado “ + ... + “# caracteres”);  
}
```

Modificacion 5

Encriptar un fichero grande es una operación lenta. Se quiere que, si un mismo fichero se va a mandar tanto por Internet como por Bluetooth, se encripte solo una vez para ganar tiempo.

```
void main () {  
    ...  
    fs.copy("privado.txt" ... "1.1.1.1" ... "Galaxy S");    // Se envía encriptado a ambos  
}
```