1. Diseñar la clase cifradorOFB que cifre bloques de 8 bits, la clave y tamaño de bloque será de 128 bits. La función de codificación realizará una XNOR y una permutación entre los distintos bloques del vector VI y de la clave K, la permutación viene expresada en la siguiente tabla.

VI15	VI14	VI13	VI12	VI11	VI10	VI9	VI8	VI7	VI6	VI5	VI4	VI3	VI2	VI1	VI0
VI3 ⊕ VI6 ⊕ K4	VI13 ⊕ VI10 ⊕ K2	VI9 ⊕ VI1 ⊕ K13	VI10 ⊕ VI14 ⊕ K7	VI12 ⊕ VI5 ⊕ K15	VI0 ⊕ VI2 ⊕ K1	VI3 ⊕ VI6 ⊕ K14	VI0 ⊕ VI13 ⊕ K6	VI8 ⊕ VI5 ⊕ K12	VI3 ⊕ VI10 ⊕ K0	VI2 ⊕ VI9 ⊕ K3	VI3 ⊕ VI11 ⊕ K11	VI1 ⊕ VI14 ⊕ K5	VI7 ⊕ VI15 ⊕ K10	VI2 ⊕ VI5 ⊕ K8	VI0 ⊕ VI4 ⊕ VI10 ⊕ K9

A este cifrador lo llamaremos OFB_Cripto. La clase tendrá un constructor al que le pasaremos la clave de 128 bits y el vector VI.

2. El objetivo de esta práctica es generar un programa que permita asegurar la autenticidad e integridad de ciertos documentos sensibles que tenemos guardados en nuestro disco duro; para ello implementaremos una función MAC (Message Authentication Code). Una de las formas de construir un algoritmo MAC es usando el cifrador simétrico de bloque en modo de operación CBC_Cripto. El documento cuyo código de integridad MAC queremos obtener se divide en bloques que ciframos a través del cifrador simétrico. Los resultados de cifrar cada uno de los bloques son procesados para obtener un código MAC. Sólo las personas que poseen la clave simétrica pueden obtener este código MAC y, por lo tanto, si un atacante modifica el documento que se quiere proteger, el propietario del documento detectará que ha habido una sustitución de los datos.

La clave se le pasa al constructor de la clase.

La clase podrá leer ficheros, que le pasemos como parámetro a uno de los métodos de la clase y devolverá el código MAC calculado. Por seguridad los códigos MAC se podrán almacenar en un fichero con el nombre y extensión que decida el usuario.

3. El algoritmo de reducción criptográfica MD5 (Message-Digest Algorithm) es un algoritmo público similar al del ejercicio anterior. Usa bloques de 128 bits y fue desarrollado por Ronald Rivest. Se puede encontrar su descripción en el RFC 1321, https://www.ietf.org/rfc/rfc1321.txt

Diseñar una clase MD5 que implemente el algoritmo, al cual se le pueda pasar un nombre de fichero y devuelva la cadena hexadecimal de 32 dígitos del código MD5

Test:

MD5("Generando un MD5 de un texto") = 5df9f63916ebf8528697b629022993e8

MD5("") = d41d8cd98f00b204e9800998ecf8427e

4. Usando PuttyGen genere una pareja de claves pública/privada de tipo SSH-1(RSA) de 4096 bits y proteja la clave privada con la frase “HolaCaracola”. El formato de la clave generada es SSH2. Usando openssl u otra herramienta similar convertir la clave pública a formato PKCS#12. Como resultado del trabajo copie los textos de la clave pública / privada y añada el fichero .pfx generado.