Criptografia
Pràctica 1

En aquesta pràctica ens familiaritzarem amb el paquet matemàtic SAGE tot implementant un criptosistema històric, concretament la xifra de Jefferson (també coneguda com a cilindre de Bazeries), una xifra inventada a finals del segle XVIII per Thomas Jefferson i reinventada de nou al segle XX durant la Segona Guerra Mundial.

Els objectius són:

- Familiaritzar-se amb l'entorn de treball SAGE.
- Implementar un criptosistema històric.

Descripció de la pràctica a realitzar

La xifra de Jefferson és una xifra (simètrica) històrica basada en l'ús d'un aparell mecànic anomenat disc de Jefferson. El disc de Jefferson és un cilindre que contenia (originalment) 36 discos, cadascun dels quals tenia les lletres de l'alfabet desordenades d'una manera diferent.

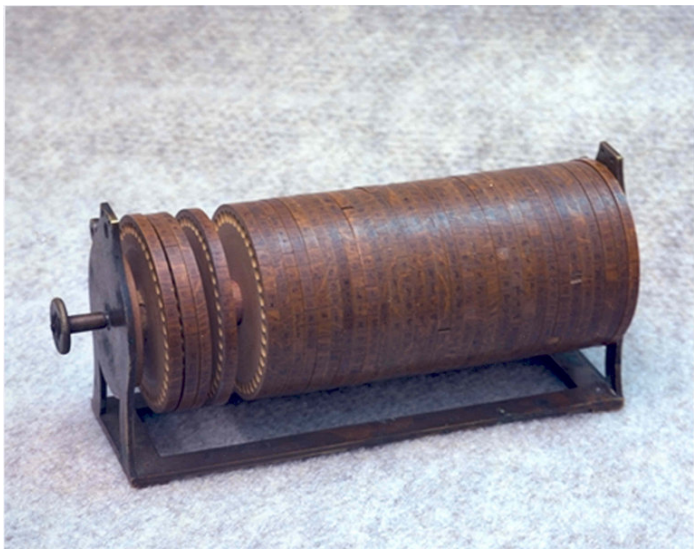

Figura 1: Disc de Jefferson (font original: wikimedia)

La clau del sistema de xifrat de la xifra de Jefferson consisteix en el conjunt de discos seleccionats (amb l'ordenació concreta de l'alfabet per cadascun d'ells) i l'ordre dels discos en el cilindre.

Una vegada situats els discos segons indica la clau, per tal de xifrar un missatge l'emissor rota els discos del cilindre fins a aconseguir alinear les 36 primeres lletres del missatge en clar. Aleshores, se selecciona una fila a l'atzar, i s'anoten les lletres de la fila com a les 36 primeres lletres del text xifrat. El procediment es pot anar repetint tantes vegades com sigui necessari per tal de xifrar tot el text que l'emissor vol enviar.

Noteu que, donada una clau i un missatge en clar, existiran múltiples textos xifrats vàlids, un per a cada fila del cilindre.

Per tal de desxifrar un text, el receptor procedeix de manera similar. En primer lloc situa els discos segons l'ordre determinat per la clau. Després, alinea els discos per tal que es pugui llegir (en horitzontal) el missatge xifrat rebut. Finalment, el receptor inspecciona totes les altres files, buscant-ne alguna que contingui un conjunt de lletres amb sentit.

En aquesta pràctica implementarem una petita generalització de la xifra de Jefferson, on tant l'alfabet dels discos com el nombre de discos del cilindre no vindran prefixats. Per implementar la xifra del disc de Jefferson dividirem la feina en tres funcions:

1. la generació de la clau
2. el xifrat d'un text en clar
3. el desxifrat d'un text xifrat

Cada un dels exercicis correspon a la programació d'una d'aquestes funcions. A continuació se'n descriuen els detalls.

Exercici 1 (3 punts)

Programació d'una funció que implementi la generació de la clau de la xifra de Jefferson

La funció podrà generar tant claus aleatòries com claus deterministes, en funció dels arguments que rebi.

La funció tindrà dos arguments opcionals, `disks` i `order`:

a) `disks`: si s'inclou l'argument `disks`, aquest contindrà una llista de mida arbitrària amb els discos a utilitzar. Cada disc vindrà especificat per una llista de caràcters (amb un alfabet també arbitrari), ordenats tal com apareixen al disc. En canvi, si l'argument `disks` s'omet (o és `None`), aleshores la funció generarà `NUM_DISKS` discos de manera aleatòria, fent servir les lletres de l'alfabet `ALPH`. Noteu que tant `NUM_DISKS` com `ALPH` són constants que venen definides a l'esquelet.

b) `order`: si s'inclou l'argument `order`, aquest contindrà una llista d'enters que especifica l'ordre en el qual se situaran els discos. En canvi, si l'argument `order` s'omet (o és `None`), la funció generarà una ordenació aleatòria dels discos.

La funció retornarà una llista amb la clau, que correspondrà al conjunt de discos ordenats, o l'error corresponent (si s'ha produït algun tipus d'error).

Nota: Recordeu validar els valors dels arguments d'entrada de la funció. Per exemple, la llista `order` no pot tenir elements repetits (ja que no es poden posar dos discos diferents en una mateixa posició). A l'esquelet de la pràctica hi trobareu les constants

`ERR_INV_DISKS` i `ERR_INV_ORDER` que cal retornar quan hi hagi, respectivament, un error amb els discos o amb l'ordre.

Exemple 1:

```
disks: ['PWQNOIZSTHBYVJEURXFCGLMKAD', 'ZDYBFSTOWUHJNAKVMEILGQPCRX',
       'CASMPHDFIXWVLNZKJBEQRTGUYO', 'TSLNXAKGPJUDCHIYEZBMFWVQOR']

order: [1, 3, 2, 0]

output: ['ZDYBFSTOWUHJNAKVMEILGQPCRX', 'TSLNXAKGPJUDCHIYEZBMFWVQOR',
'CASMPHDFIXWVLNZKJBEQRTGUYO', 'PWQNOIZSTHBYVJEURXFCGLMKAD']
```

Exemple 2:

```
disks: None

order: None

output: ['MLUWKQGOVEBDXTJCHRIZPNFASY', 'YEZOSRGAMNVCPHWFUQJTIBLDKX',
'PWMYHSGIOAREZDBUJLXVNCFKTQ', 'AESDWLNQUBHIMOJZKRTCGYVPFX', 'YAOXEHTKWVINSCQJMDPUFZRLBG',
'GLRZXFUBPDMJCATHIYVWENKQSO',

# ... llista de NUM_DISKS elements

'WKBXYCSFHQDOPZIERUMTJGAVLN', 'NYHDMCWSAGQXZBLTIRVUKPFJEO']
```

Exercici 2 (3 punts)

Programació d'una funció que implementi el procés de xifrat de la xifra de Jefferson

La funció prendrà com a arguments a) la clau i b) el missatge.

a) `key`: el primer argument contindrà una llista amb la clau, tal com s'obté de la funció de generació de la clau.

c) `message`: el segon argument contindrà una cadena de caràcters amb el missatge de text en clar que volem xifrar. Podeu assumir que el text en clar sempre es podrà xifrar (l'alfabet utilitzat per a escriure el text en pla és el mateix que el que s'ha fet servir per generar la clau) i que la mida del missatge sempre serà inferior al número de discos del cilindre.

La funció retornarà una cadena de caràcters amb un dels possibles missatges xifrats, seleccionat de manera aleatòria d'entre tots els possibles textos xifrats que corresponguin al text en clar.

Exemple:

```
key: ['RAMBOUZSXQJVKNEYWPLCITDFHG','DXFIJGWKHAONPTCSUVBMLYREQZ',
       'AOYRZHSTDVGXJFCEKIUNWQMPBL', 'BQILVSENZRCGYFJWDPOUMHAKXT',
       'PGZTLMRYJSQIDBXUWEHVFOACKN', 'HMINVOKABDJSFPXECURLTWQZGY']

message: 'CRIPTO'

output: 'UHRLUU'
```

Noteu que la sortida no és única (ja que el xifrat és probabilístic).

Exercici 3 (4 punts)

Programació d'una funció que implementi el procés de desxifrat de la xifra de Jefferson

La funció prendrà com a arguments a) la clau i b) el missatge xifrat.

a) `key`: el primer argument contindrà una llista amb la clau, tal com s'obté de la funció de generació de la clau.

b) `message`: el segon argument contindrà el missatge xifrat que volem desxifrar.

La funció retornarà el missatge en clar, si algun dels possibles textos en clar corresponents al text xifrat es troben en el diccionari de paraules reconegudes (emmagatzemat a la constant `DICTIONARY_WORDS`), o bé una llista amb tots els possibles missatges en clar en cas contrari.

Exemple 1:

```
key: ['RAMBOUZSXQJVKNEYWPLCITDFHG','DXFIJGWKHAONPTCSUVBMLYREQZ',
      'AOYRZHSTDVGXJFCEKIUNWQMPBL', 'BQILVSENZRCGYFJWDPOUMHAKXT',
      'PGZTLMRYJSQIDBXUWEHVFOACKN', 'HMINVOKABDJSFPXECURLTWQZGY']

message: 'UHRLUU'

output: 'CRIPTO'
```

Exemple 2:

```
key: ['RAMBOUZSXQJVKNEYWPLCITDFHG','DXFIJGWKHAONPTCSUVBMLYREQZ',
      'AOYRZHSTDVGXJFCEKIUNWQMPBL', 'BQILVSENZRCGYFJWDPOUMHAKXT',
      'PGZTLMRYJSQIDBXUWEHVFOACKN', 'HMINVOKABDJSFPXECURLTWQZGY']

message: 'NPCOQ'

output: ['NPCOQ', 'ETEUI', 'YCKMD', 'WSIHB', 'PUUAX', 'LVNKU', 'CBWXW', 'IMQTE', 'TLMBH',
'DYPQV', 'FRBIF', 'HELLO', 'GQAVA', 'RZOSC',  'ADYEK', 'MXRNN', 'BFZZP', 'OIHRG', 'UJSCZ',
'ZGTGT', 'SWDYL',   'XKVFM', 'QHGJR', 'JAXWY', 'VOJDJ', 'KNFPS']
```

Nota: La cadena `'NPCOQ'` és el resultat de xifrar la paraula `'HELLO'`, que no es troba al diccionari de paraules reconegudes.

Format i data de lliurament

La data màxima de lliurament de la pràctica és el 16/10/2017 (a les 24 hores).

Juntament amb l'enunciat de la pràctica trobareu l'esquelet de la mateixa en format SAGE notebook worksheet (extensió .sws). Aquest mateix fitxer és el que heu de lliurar un cop hi codifiqueu totes les funcions.

En aquest esquelet també hi trobareu inclosos els jocs de proves dels diferents apartats. Tal com s'esmenta en el fitxer sws, **no es pot modificar cap part del fitxer corresponent al joc de proves**.

El lliurament de la pràctica constarà de d'**un únic fitxer** SAGE notebook (extensió sws) o bé sage cloud (extensió sagews) on hagueu inclòs la vostra implementació.