Universidad Nacional Autónoma de México
Facultad de Ciencias - Ciencias de la Computación
Criptografía y seguridad 2019-2

# Ejercicio 2

Haz un programa que automatice el criptoanálisis del esquema de Vigenère. El programa recibirá un texto cifrado y devolverá el texto original (sin tener la clave). Ya que el método es solo una heurística, la salida en realidad serán varios textos, donde *se espera* encontrar el texto claro correcto.

El alfabeto serán las 27 letras mayúsculas del español

`ABCDEFGHIJKLMNÑOPQRSTUVWXYZ`

Si en el texto cifrado aparecen espacios o símbolos como puntos o comas, puedes omitirlos totalmente o puedes omitirlos para el criptoanálisis y recuperarlos una vez que obtengas el texto claro. Por ejemplo, el criptotexto `QHEK, UTÑH NXQMS` debes tratarlo como `QHEKUTÑHNXQMS`, y ya que recuperes el original `HOLACOMOESTAS` opcionalmente puedes agregar los símbolos y espacios (en sus posiciones originales) para obtener el original `HOLA, COMO ESTAS`.

**Programa**

El programa se llamará `vigenere` y aceptará un parámetro para indicar el archivo cifrado. Se debe ejecutar de la siguiente manera

```
$ python vigenere.py textocifrado
```

El texto cifrado tendrá codificación UTF-8 (para poder usar la Ñ).

Como salida se mostrarán en pantalla tres textos claros con la respectiva clave que fue usada para obtenerlos. Usarás los tres mejores valores del tamaño de clave $t$. De cada texto solo se mostrarán los primeros 50 caracteres.

Ejemplo de salida:

```
1. Clave: GHCJ
Texto: KJNK KJ NK JN KJÑQC JKJNNKJM, KJNA DS ASDAD ASD ASÑAD SAVZZC

2. Clave: PERRITO
Texto: ESTE SI ES EL TEXTO CORRECTO, ERES EL MEJOR DEL MUNDO MUNDIA

3. Clave: INCORRECTA
Texto: JDND TX DS QW ÑVOJF DJDKDPDI, NVJV ZX MVNDD QPA EROQI UUBTCN
```

Recuerda que puedes omitir los espacios y símbolos en la salida. Además no siempre se encontrará la clave correcta, pero con suerte puede aparecer una clave *casi* correcta.

**Cómo obtener el tamaño de la clave**

Como entrada se tiene el criptotexto $C = c_0, c_1, \ldots, c_n$.
Para $t = 1, 2, \ldots$ haz lo siguiente:

1. Forma el bloque $B_0 = c_0, c_{0+t}, c_{0+2t}, \ldots$
2. Para $i = 1, \ldots, 27$ obtén la frecuencia $q_i$ de la $i$-ésima letra sobre el bloque $B_0$, es decir, calcula el porcentaje de aparición de cada letra del alfabeto. Tendrás $q_1, \ldots, q_{27}$ tales que $\sum q_i = 1$.
3. Calcula $I = \sum_{i=1}^{27} q_i^2$ y compáralo con el valor 0.0741.

Con alta probabilidad, el primer valor de $t$ que aproxime $I$ a 0.0741 lo suficiente será el tamaño de la clave (tú definirás qué tanto es suficiente). Nota que un valor de $t$ incorrecto (y que no sea múltiplo del valor correcto) hará que se obtenga un $I$ más cercano a 0.037.

Para comprobar si el valor de $t$ es correcto puedes hacer el procedimiento anterior con otros bloques $B_i$ en vez de $B_0$.

**Cómo descifrar cada bloque**

Una vez que conocemos el valor de $t$, podemos formar los bloques $B_i = c_i, c_{i+t}, c_{i+2t}, \ldots$ y aplicarle a cada uno el criptoanálisis de César. Como el texto claro de cada bloque no tendrá significado en español, para automatizar el procedimiento hay que hacer lo siguiente.

Sea $p_i$ la frecuencia de la $i$-ésima letra en textos normales en español, y sea $B$ el bloque que se quiere descifrar.

1. Obtén las frecuencias $q_i$ en el texto $B$, como se describió en el paso 2 más arriba.
2. Para $k = 0, \ldots, 26$ obtén los valores

$$I_k = \sum_{i=1}^{27} p_i \cdot q_{i+k} \pmod{27}$$

El valor $I_k$ más cercano a 0.0741 es el que nos dará el desplazamiento $k$ para descifrar el bloque.

## Entrega

Organiza tus archivos en un archivo zip con nombre `Ejerc2_[Ape Paterno]_[Ape Materno].zip` y súbelo al Classroom. No subas archivos sueltos, solo el archivo zip.

Fecha límite: 20 de febrero.