Trabalho 1
Cifra de Vigenère

Paulo Victor França de Souza - 20/0042548
Thais Fernanda de Castro Garcia - 20/0043722

[1]Dep. Ciência da Computação – Universidade de Brasília (UnB)
CIC0201 - Segurança Computacional (2022.1)
Prof. João José Costa Gondim - Turma 1

1. Introdução

Sendo um método de criptografia, a cifra de Vigenère é uma forma simples de substituição polialfabética e utiliza uma série de diferentes cifras de César com diferentes transformações, onde a sequência é definida por palavra-chave, onde cada letra define a mudança necessária.

Neste descritivo serão retratados detalhes acerca da cifra de Vigenère, sendo feito na linguagem de programação python a implementação de um cifrador e decifrador, e também, uma forma de ataque de recuperação de senha por análise de frequência.

Para o cifrador funcionar, é necessário receber uma senha e uma mensagem que será cifrada com base na cifra de Vigenère e ao final será gerado um criptograma, ou seja, o texto cifrado. No caso do decifrador, para seu funcionamento, ele recebe uma senha e um criptograma que é decifrado com base na cifra de Vigenère, obtendo a recuperação de uma mensagem. No caso do ataque de recuperação de senha, para seu funcionamento é necessário receber a mensagem cifrada, podendo ser em português ou inglês, e por meio de uma série de processos ao final é recuperada a senha geradora do keystream usado na cifração, e consequentemente, obtida a decifração do texto. [3] [2]

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Figure 1. Grade de Vigenère.

2. Implementação

2.1. Parte I: Cifrador/Decifrador

Primeiramente foi definido o alfabeto, o qual pode ser alterado conforme necessidade, com suas 26 letras unificadas em formato de string (”ABCDEFGHIJKLMNOPQRSTUVWXYZ”) e criado dois dicionários chamados *letra_para_numero* e *numero_para_letra*, onde o primeiro tem como chave caracteres o segundo tem como chave números. Para maior facilidade de cifração e evitar erros, a mensagem foi tratada de uma forma a ficar mais simples, isto é, apenas com caracteres maiúsculos, sem acentuação, pontuação, números e espaços.

Para ser feita decifração da mensagem (função *cifra()*), há a iteração da lista de listas e iterado novamente letra por letra das listas com tamanho igual ao tamanho da chave, criando gradualmente a mensagem cifrada, sendo acrescentado na variável *mensagem_cifrada* a letra correspondente a operação de cifração de Vigenère (Pi + Ki) % tamanho do alfabeto, onde Pi é o valor numérico de uma letra da mensagem a ser cifrada, e Ki é o valor numérico da chave com índice equivalente.

Para ser feita a decifração da mensagem (função *decifra()*), há a realização de etapas parecidas com a de cifração, utilizando como operação (Pi - Ki) % tamanho do alfabeto, ou seja, é feito a operação inversa e obtida a decifração da mensagem, no entanto a função verifica se os caracteres pertencentes à mensagem pertencem também ao alfabeto, ignorando caso não.

```python
def cifra(mensagem, chave): # Cifração com base na cifra de vigenere.
    mensagem_cifrada = []

    for i,c in enumerate(mensagem):
        numero = (letra_para_numero[c] + letra_para_numero[chave[i]]) % len(alfabeto) # Operação (Pi + Ki) % tamanho do alfabeto
        mensagem_cifrada.append(numero_para_letra[numero]) # Converte os números para letras
    return("".join(mensagem_cifrada))
```

Figure 2. Cifração.

```python
def decifra(mensagem_cifrada, chave, alfabeto): # Decifração com base na cifra de vigenere.
    resultado = []
    pulos = 0
    for i,c in enumerate(mensagem_cifrada):
        if c not in alfabeto:
            resultado.append(c)
            pulos += 1
            continue

        numero = (letra_para_numero[c] - letra_para_numero[chave[(i - pulos) % len(chave)]]) % len(alfabeto) # Operação (Pi - Ki) % tamanho do alfabeto
        resultado.append(numero_para_letra[numero]) # Converte os números para letras
    return("" . join(resultado))
```

Figure 3. Decifração.

2.2. Parte II: Ataque de Recuperação de Senha Por Análise de Frequência Baseado no Exame de Kasinski

Para ser feito o ataque de recuperação de senha, primeiramente são definidos dois dicionários de frequência, um em inglês (*frequencia_ingles*) e outro em português (*frequencia_portugues*), isto é, cada um possui a porcentagem da frequência que cada letra do alfabeto é utilizada no respectivo idioma. Cada dicionário tem como chave um caractere ligado a um valor que representa sua frequência no inglês ou português, respectivamente. Em seguida começa o processo para encontrar um tamanho aproximado para

a chave usada na criptografia de uma frase, onde por meio de uma lista de espaçamento que para evitar erros possui um limite do tamanho da provável chave, que é configurável. Em seguida é realizada o agrupamento do código cifrado em grupos com 3 caracteres (trigramas), é preenchida a lista de espaçamento apenas com o valor de suas distâncias, sendo criado uma lista de espaçamentos. Em seguida, é iniciado o processo final desse procedimento, tendo como base a lista de espaçamento para ser feito um cálculo de qual seria o valor de maior incidência na lista e colocado como possível tamanho da chave, foi utilizada uma margem de erro de 20% para abranger casos onde as dois tamanhos de chave mais prováveis, dando preferência para o maior caso. Para concluir o ataque de recuperação de senha, é gerado um dicionário de frequência para cada letra possível naquela posição da chave, com isso é comparado essas frequências com a frequência real do alfabeto e selecionado a que mais se aproxima para essa comparação. É utilizado a posição dos picos de frequência do alfabeto, para isso optamos, de forma arbitrária utilizamos um numero de 5 topos. [4] [1]

```
frequencia_ingles = [('A', 8.167), ('B', 1.492), ('C', 2.782), ('D', 4.253),
            ('E', 12.702), ('F', 2.228), ('G', 2.015), ('H', 6.094),
            ('I', 6.966), ('J', 0.153), ('K', 0.772), ('L', 4.025),
            ('M', 2.406), ('N', 6.749), ('O', 7.507), ('P', 1.929),
            ('Q', 0.095), ('R', 5.987), ('S', 6.327), ('T', 9.056),
            ('U', 2.758), ('V', 0.978), ('W', 2.360), ('X', 0.150),
            ('Y', 1.974), ('Z', 0.074)]
frequencia_portugues = [('A', 14.63), ('B', 1.04), ('C', 3.88), ('D', 4.99),
            ('E', 12.57), ('F', 1.02), ('G', 1.30), ('H', 1.28),
            ('I', 6.18), ('J', 0.40), ('K', 0.02), ('L', 2.78),
            ('M', 4.74), ('N', 5.05), ('O',10.73), ('P', 2.52),
            ('Q', 1.20), ('R', 6.53), ('S', 7.81), ('T', 4.34),
            ('U', 4.63), ('V', 1.67), ('W', 0.01), ('X', 0.47),
            ('Y', 0.01), ('Z', 0.47)]
```

Figure 4. Frequências das letras em inglês e português.

References

[1] R. Daniel. Kasiski analysis: Breaking the code.

[2] PlanetCalc. Cifra de vigenère.

[3] Wikipédia, a enciclopédia livre. Cifra de vigenère.

[4] Wikipédia, a enciclopédia livre. Frequência de letras.