Segurança Computacional
Cifra de Vignère

Pedro Henrique de Brito Agnes, 18/0026305
Dep. Ciência da Computação - Universidade de Brasília (UnB)

1 Cifrador/Decifrador

Para a implementação do cifrador e decifrador da cifra de Vigenère foi criado um programa em python na versão 3.6, que está no arquivo `src/parte1.py`. Na execução, o programa aceita alguns parâmetros, porém pode ser executado sem nenhum da seguinte forma:

```
1 python src/parte1.py        # cifrador
2 python src/parte1.py -d     # decifrador
```

Pela abordagem acima, será solicitada durante a execução, uma mensagem/criptograma e uma chave e terá como saída a mensagem cifrada/decifrada de acordo com a cifra de Vigenère e com a chave informada. Conforme informado, o programa aceita certos parâmetros. Para visualizar a lista completa e o que fazem, pode ser utilizada a flag `-h` ou `--help`:

```
1 python src/parte1.py -h
```

O primeiro parâmetro que pode ser passado é um arquivo que conterá a mensagem seguido do segundo que também é um arquivo onde será passada a chave. Opcionalmente, pode-se passar um arquivo de saída com a flag `-o` e, caso não seja passada, a saída (mensagem cifrada/decifrada) será printada no terminal.

Para cifrar uma mensagem, pode ser executado o comando abaixo como exemplo:

```
1 python src/parte1.py sample/msg.txt sample/key.txt -o sample/out.txt
```

Onde `sample/msg.txt` é a mensagem de exemplo no diretório `sample`, `sample/key.txt` é a chave e `sample/out.txt` é o arquivo onde será impressa a saída. Da mesma forma, para decifrar a mensagem cifrada com o comando anterior, pode ser usado o comando a seguir:

```
1 python src/parte1.py sample/out.txt sample/key.txt -o
    sample/decifrada.txt -d
```

A flag `-d` ao final é que indica ao programa que a mensagem será decifrada e não cifrada. No exemplo, a saída foi definida para aparecer no arquivo `sample/decifrada.txt`.

1.1 Aspectos Técnicos

Toda a lógica utilizada para a implementação do cifrador/decifrador estão em um arquivo separado `src/vigenere/cipher.py`. Este arquivo tem uma função principal `cifra` que recebe uma mensagem, uma chave e uma variável booleana `opt`, que define se a mensagem deve ser cifrada (`false`) ou decifrada (`true`). Para as operações, é basicamente feita uma soma dos valores ascii da letra atual da mensagem subtraída do ascii da letra 'a' com o ascii da letra atual da *keystream*. Com o valor obtido na operação anterior, é realizado o módulo por 26 (quantidade de letras no alfabeto) e assim, é somado novamente o valor do ascii de 'a' para obter o caractere do criptograma. Este processo é feito em loop até que se chegue ao final da mensagem, porém caracteres que não sejam letras ou acentuados não são processados e se mantém inalterados no output. Já a decifração funciona de forma muito similar à citada acima, mudando apenas a operação de soma para subtração entre o ascii da letra da mensagem e da *keystream*.