Cifra por Deslocamento

1. Força Bruta

Na força bruta, testamos todos os deslocamentos possíveis (geralmente entre 1 e 25 em uma cifra de César, já que o alfabeto tem 26 letras). Para cada tentativa, decifamos o texto e verificamos se ele faz sentido.

- **Viabilidade**: A força bruta é viável para cifras de deslocamento, especialmente com textos curtos, devido ao número pequeno de possibilidades. Contudo, se aplicássemos a mesma técnica para cifras mais complexas, como a Vigenère com uma chave longa, a abordagem se tornaria inviável.
- **Complexidade e Tempo de Execução**: A complexidade é O(n·m)O(n \cdot m)O(n·m), onde nnn é o tamanho do texto e mmm o número de possíveis deslocamentos (para a cifra de César, m=25m = 25m=25). Para cifras curtas, a execução é rápida, mas o tempo aumenta com o comprimento do texto e o número de deslocamentos possíveis. Na prática, o algoritmo é linear em relação ao número de tentativas.

2. Análise de Frequência

A análise de frequência explora as características estatísticas da língua para decodificar o texto, comparando a frequência de letras do texto cifrado com frequências esperadas. Em português, por exemplo, letras como “a”, “e” e “o” aparecem mais frequentemente.

- **Viabilidade**: Esta técnica é mais robusta e aplicável a cifras mais complexas, incluindo cifras de substituição monoalfabética. Em cifras de César, a análise de frequência pode identificar o deslocamento sem testar todas as possibilidades, especialmente em textos mais longos, onde as distribuições de frequência se estabilizam.
- **Complexidade e Tempo de Execução**: O processo é mais complexo que a força bruta, mas eficiente para cifras de deslocamento e substituição. Sua complexidade depende da análise das frequências das letras no texto cifrado e da comparação com as frequências da língua, tornando-a O(n)O(n)O(n) no comprimento do texto. Em geral, o tempo de execução é menor do que o de força bruta, pois não exige a tentativa de todos os deslocamentos possíveis.

Cifra por Transposição

1.Estratégia de cifra

- **Remoção de espaços e preparação da mensagem:** A primeira etapa é remover os espaços da mensagem, garantindo que ela seja processada sem separações. Isso é importante para garantir que a mensagem cifrada mantenha o mesmo comprimento que a original, sem adicionar espaços extras que possam interferir no processo de transposição.

- **Construção da matriz:** A mensagem é então organizada em uma matriz, onde:

    - O número de colunas da matriz é determinado pelo comprimento da chave (`num_colunas`).
    - O número de linhas (`num_linhas`) é calculado dividindo o comprimento da mensagem pelo número de colunas. Se houver sobra (mensagem não é divisível por `num_colunas`), uma linha extra é adicionada.

As células da matriz são preenchidas com os caracteres da mensagem, linha por linha. Caso a mensagem não preencha a matriz completamente, as células vazias são preenchidas com espaços em branco.

- **Permutação das colunas:** O passo crucial da cifra de transposição é a permutação das colunas com base na chave fornecida. A chave define a ordem em que as colunas da matriz serão lidas:

    - Cada número na chave representa um índice de coluna. Por exemplo, se a chave for `[3, 1, 2]`, as colunas serão lidas na ordem da coluna 3, seguida pela coluna 1, e por último pela coluna 2.
    - Para isso, a matriz é lida de acordo com a ordem definida pela chave, criando novas colunas (armazenadas em `matriz_permutada`) com os caracteres reorganizados.

- **Construção da mensagem cifrada:** Finalmente, as colunas permutadas são concatenadas para formar a mensagem cifrada. O resultado é uma string que representa a mensagem original, mas com os caracteres reorganizados de acordo com a chave de transposição.

2.Estratégia de quebra da cifra

- **Escolha da chave:** A primeira etapa envolve escolher o tamanho máximo de chave que será testado, para esse passo é recomendo a chave ter no máximo o mesmo tamanho do texto cifrado, porem por questões de limitações computacionais nem sempre isso será possível
- **Tentativa de quebra:** A estratégia de quebra usa uma combinação de força bruta e análise de frequência:
    - Primeiramente testamos todas as possibilidades até chegar no tamanho máximo de chave, declarado na primeira etapa

- A medida que os testes ocorrem combinações que contém uma lista de palavras previamente declaradas, escolhidas a partir de analises de frequência, são apresentadas para o usuário
- Com as combinações mais promissoras o usuário pode então analisar qual faz mais sentido no contexto
- Obs: embora o código fonte compare todas as combinações com a variável “resposta” em uma aplicação prática essa variável não existiria, ela só foi incluída para que seja possível confirmar que a função eventualmente seria capaz de a resposta correta.