Tutorial: Decodificando Mensagens

Arthur Andrade D'Olival

O problema pode ser resolvido utilizando uma abordagem de **programação dinâmica**, na qual calculamos, passo a passo, o número de formas possíveis de decodificar a sequência até cada posição.

Modelagem

Cada dígito (ou par de dígitos consecutivos) da sequência pode representar uma letra do alfabeto latino, seguindo a regra:

$$1 \to A, \quad 2 \to B, \quad \dots, \quad 26 \to Z$$

Portanto, precisamos contar todas as maneiras válidas de interpretar a sequência numérica, considerando que apenas combinações entre 1 e 26 são válidas.

Definição da DP

Definimos uma estrutura de DP com dois estados para cada posição i:

- $dp[i][0]$: número de mensagens possíveis terminando no caractere i **quando o dígito atual não é concatenado** com o anterior;
- $dp[i][1]$: número de mensagens possíveis terminando no caractere i **quando o dígito atual é concatenado** com o anterior.

Função de Transição

A função de transição é definida da seguinte forma:

$$dp[i][0] = \begin{cases} dp[i-1][0] + dp[i-1][1], & \text{se } 1 \leq s[i] \leq 9 \\ 0, & \text{caso contrário} \end{cases}$$

$$dp[i][1] = \begin{cases} dp[i-2][0] + dp[i-2][1], & \text{se } 10 \leq 10 \cdot s[i-1] + s[i] \leq 26 \\ 0, & \text{caso contrário} \end{cases}$$

Casos Base

Os casos base são:

$$dp[0][0] = 1, \quad dp[0][1] = 0, \quad dp[1][0] = 1, \quad dp[1][1] = 0$$

Isso ocorre porque, antes de processar qualquer caractere, há exatamente uma maneira "vazia" de formar uma mensagem válida, e o primeiro dígito pode, no máximo, representar uma única letra isolada.

Resposta Final

O total de formas possíveis de decodificar a sequência é dado por:

$$dp[n][0] + dp[n][1]$$

onde n é o comprimento da sequência de entrada.

Complexidade

- **Tempo:** $O(n)$, pois cada caractere é processado uma única vez;
- **Espaço:** $O(n)$, podendo ser otimizado para $O(1)$ ao armazenar apenas os últimos dois estados.