

# Tutorial: Conversor Fonético Genérico

## Leetcode 127

O problema de transformação de palavras exige encontrar o menor caminho entre uma palavra inicial e uma palavra final, alterando apenas uma letra por vez, com a restrição de que todas as palavras intermediárias devem pertencer a um dicionário fornecido.

### Representação do estado

- Cada palavra do dicionário (incluindo a palavra inicial) representa um nó em um grafo não direcionado.
- Uma aresta conecta duas palavras se elas diferem em exatamente uma letra, uma "mutação" válida.
- A transformação mais eficiente equivale a encontrar o caminho mais curto entre o nó inicial e o nó final neste grafo.

### Estratégia de solução usando BFS (Busca em Largura)

Como precisamos encontrar o menor número de passos em um grafo com arestas de peso uniforme e não direcionadas, a Busca em Largura (BFS) é a abordagem algorítmica ideal.

1. Leia o número de palavras no dicionário ( $N$ ) e o comprimento das palavras ( $M$ ), seguidos das palavras inicial e final.
2. Armazene todas as palavras do dicionário em um conjunto (*hash set*) para buscas eficientes. Insira também a palavra inicial neste conjunto, pois ela é o nosso ponto de partida.
3. Construa o grafo através de uma lista de adjacências:
  - Para cada palavra no conjunto, itere sobre cada um de seus  $M$  caracteres.
  - Substitua o caractere atual temporariamente por todas as 26 letras minúsculas do alfabeto ('a' a 'z').
  - Se a nova palavra gerada após essa substituição também existir no dicionário, adicione-a à lista de adjacências da palavra original.
4. Inicialize a simulação da BFS: crie uma fila e insira a palavra inicial. Utilize um conjunto auxiliar para registrar os nós já visitados (**seen**), evitando ciclos infinitos. Defina a distância (profundidade) inicial da sequência como 1.
5. Processe a fila operando por camadas (nível por nível):
  - Descubra quantos elementos existem na fila atual. Este é o número de palavras no nível de profundidade atual.
  - Para cada uma dessas palavras, retire-a da fila e verifique: se for a palavra final desejada, a missão foi concluída! Imprima a profundidade atual e encerre.
  - Caso contrário, acesse a lista de adjacência da palavra. Para cada vizinho (mutação válida) que ainda não foi visitado, insira-o na fila e marque-o como visitado.

6. Ao terminar de processar todas as palavras da camada atual, incremente a profundidade.
7. Se a fila se esvaziar completamente e a palavra final nunca for atingida, a transformação é impossível sob essas regras. Retorne 0.