## **Processamento de dados brutos de testes**

Este script foi desenvolvido para **processar dados brutos de testes**, com os seguintes objetivos:

* Calcular o número mínimo de movimentos necessários para resolver cada puzzle.
* Determinar a pontuação acumulada, refletindo a eficiência na resolução.

**FUNCIONALIDADE PRINCIPAL:**

O script percorre todos os arquivos CSV localizados na pasta **'dados\_originais'**. Para cada arquivo, executa as seguintes ações:

* Lê o histórico de movimentos registrados.
* Identifica os estados inicial e final do puzzle.
* Calcula a solução ótima, ou seja, o menor número possível de movimentos.
* Compara os movimentos realizados pelo participante com a solução ótima.
* Calcula a pontuação do trial, aplicando penalidades por movimentos excedentes.
* Gera uma nova coluna com a pontuação acumulada.

**CONCEITO:**

O puzzle baseia-se na lógica da Torre de Londres, envolvendo pinos e bolas.

* Cada estado do puzzle é representado como uma string no formato '|A|BC| |'.
* O script padroniza a estrutura para sempre incluir 3 pinos.
* Um algoritmo de busca em largura (BFS) é empregado para encontrar a solução mais curta.
* Se a solução for considerada impossível (estado inválido), o trial recebe 0 pontos.

**ETAPAS DE PROCESSAMENTO:**

1. **Conversão de Estado:**
   * Converte as strings de estado para uma estrutura de listas aninhadas.
   * Normaliza os estados para garantir consistência (sempre 3 pinos).
2. **Cálculo de Movimentos Mínimos:**
   * Utiliza busca em largura para identificar a sequência mínima de passos.
   * Valida as restrições de altura máxima para cada pino.
3. **Pontuação:**
   * A pontuação inicial é de 10 pontos por trial concluído com sucesso.
   * Deduz-se 1 ponto para cada movimento extra além do mínimo necessário.
   * A pontuação não pode ser negativa.
   * Trials não concluídos ('done' != 1) não geram pontuação.
4. **Saída:**
   * Adiciona as colunas 'movimentos\_minimos' e 'pontuacao\_acumulada' ao arquivo CSV.
   * Salva o arquivo processado na pasta 'resultados\_processados', mantendo o nome original.

**TRATAMENTO DE EXCEÇÕES:**

* O script detecta estados inválidos e implementa limites para evitar loops infinitos.
* Em caso de erro em uma linha específica, os 'movimentos\_minimos' são marcados com '-1' e a pontuação com '0'.

**LOGS:**

* Gera logs detalhados de cada etapa do processamento, incluindo estados, solução mínima e pontuação.
* Exibe mensagens de erro para indicar problemas em arquivos ou linhas específicas.

**MODO DE USO:**

1. Coloque os arquivos CSV originais na pasta **`dados\_originais`**.
2. Execute o script com o comando: **`python process\_all\_files.py`**
3. Os arquivos processados serão salvos automaticamente na pasta **`resultados\_processados`**.