

# Problema de Satisfação de Restrições - N-Rainhas

## 1. Descrição

O projeto implementa o algoritmo de **Busca por Backtracking** para resolver um **Problema de Satisfação de Restrições (CSP)**. O problema modelo utilizado é o clássico **N-Rainhas**, que consiste em posicionar N rainhas de xadrez em um tabuleiro N×N de forma que nenhuma rainha possa atacar a outra.

---

## 2. Modelagem (como CSP)

A formulação do problema como um CSP foi a seguinte:

- Variáveis:**  $V = \{C_0, C_1, \dots, C_{N-1}\}$  (Uma variável para cada coluna do tabuleiro).
  - Domínio:**  $D = \{0, 1, \dots, N - 1\}$  (O valor de cada variável  $C_i$  é a linha onde a rainha da coluna  $i$  será posicionada).
  - Restrições:** Para quaisquer duas rainhas  $(C_i, \text{linha}_i)$  e  $(C_j, \text{linha}_j)$  com  $i \neq j$ :
    - Restrição de Linha:**  $\text{linha}_i \neq \text{linha}_j$ .
    - Restrição de Diagonal:**  $|\text{linha}_i - \text{linha}_j| \neq |i - j|$ .
  - Nota: A restrição de coluna é satisfeita implicitamente pela modelagem, pois cada variável representa uma coluna.*
- 

## 3. Algoritmo

- Busca por Backtracking:** O algoritmo resolve o problema de forma recursiva, atribuindo um valor (linha) a uma variável (coluna) por vez.
    - Tenta posicionar uma rainha na próxima coluna.
    - Verifica se a posição viola as restrições com as rainhas já posicionadas.
    - Se a posição é **válida**, avança recursivamente para a próxima coluna.
    - Se a posição é **inválida** (ou se a chamada recursiva falha), o algoritmo **retrocede (backtrack)**, desfaz a atribuição e tenta o próximo valor (próxima linha) no domínio.
    - O processo termina quando uma solução é encontrada (todas as colunas preenchidas) ou quando todas as possibilidades foram exaustivamente testadas sem sucesso.
-

## 4. Resultados (para N=8)

O programa foi executado para o problema clássico de 8-Rainhas ( N=8 ).

- **Solução Encontrada:** [(0, 0), (1, 4), (2, 7), (3, 5), (4, 2), (5, 6), (6, 1), (7, 3)]
- **Nós Visitados:** 114
- **Tempo de Execução:** 0.88 ms

O algoritmo encontrou uma solução válida, conforme visualizado abaixo:

```
Q . . . . . . .  
. . . . . Q .  
. . . . Q . . .  
. . . . . . Q  
. Q . . . . .  
. . . Q . . . .  
. . . . . Q . .  
. . Q . . . . .
```

---

## 5. Conclusão

O algoritmo de **Backtracking** mostrou-se extremamente eficaz para resolver o problema das N-Rainhas. Ao verificar as restrições a cada passo, o algoritmo "poda" vastas seções da árvore de busca, evitando a necessidade de explorar o espaço de estados completo (que para N=8 seria  $8^8 \approx 16.7$  milhões de estados). A execução encontrou uma solução visitando apenas **114 estados**, demonstrando a eficiência do CSP como uma técnica poderosa de resolução de problemas baseada em restrições.