

8-QUEENS Busca Gulosa

- Função inicial para chamar a geração de todas as soluções possíveis para o problema

```
In [10]: class NQueens:
    def __init__(self, tamanho):
        # Os vetores abaixo salvam soluções encontradas
        self.tamanho = tamanho
        self.solucoes = 0
        self.fSolucao()
```

Função Resolução

- Função para inserir rainhas no tabuleiro de acordo com soluções encontradas e chamar para imprimir as soluções encontradas

```
In [2]: def fSolucao(self):

    posicoes = [-1] * self.tamanho
    self.insererRainha(posicoes, 0)
    print("Show! Encontramos o total de:", self.solucoes, "soluções.")
```

Função de Inserção Rainha

- Função gulosa que tenta inserir uma rainha em uma posição checando N possibilidades;
- Se uma posição valida for encontrada, chama a própria função e tenta inserir uma rainha;
- Se não encontrar, segue a próxima tupla (linha) até inserir as N rainhas

```
In [3]: def insererRainha(self, posicoes, alvoLinha):

    # Critério de Parada: Até que todas as N rainhas sejam posicionadas
    if alvoLinha == self.tamanho:
        self.exibeTabuleiro(posicoes)
        #self.exibeTabuleiroSimples(posicoes)
        self.solucoes += 1
    else:
        # Percorrendo as N colunas para posicionar (força bruta)
        for coluna in range(self.tamanho):
            # Verificando Lugares ocupados e inserindo rainhas
            if self.checaPosicao(posicoes, alvoLinha, coluna):
                posicoes[alvoLinha] = coluna
                self.insererRainha(posicoes, alvoLinha + 1)
```

Função de Checagem de Ataque

- Função para checar se uma rainha posicionada pode estar sobre ataque;
- Checa-se a última rainha posicionada -1 (coluna e posições na diagonal)

```
In [14]: def checaPosicao(self, posicoes, posiOcupada, coluna):

    for i in range(posiOcupada):
```

```

        if posicoes[i] == coluna or \
           posicoes[i] - i == coluna - posiOcupada or \
           posicoes[i] + i == coluna + posiOcupada:
            return False
    return True

```

Função Tabuleiro

- Função para imprimir o tabuleiro com as posições das rainhas

```

In [15]: def exibeTabuleiro(self, posicoes):

        for tupla in range(self.tamanho):
            linha = ""
            for coluna in range(self.tamanho):
                if posicoes[tupla] == coluna:
                    linha += "R "
                else:
                    linha += ". "
            print(linha)
        print("\n")

```

Função Tabuleiro (curta)

- Imprime o tabuleiro de forma simplificada, cada número representa uma posição ocupada correspondente a uma linha

```

In [16]: def exibeTabuleiroSimples(self, posicoes):

        linha = ""
        for i in range(self.tamanho):
            linha += str(posicoes[i]) + " "
        print(linha)

```

Função Main

- Função Main para inicializar o tabuleiro e chamar demais funções

```

In [17]: def main():
        NQueens(8)
        main()

```

N-QUEENS -> Script

```

In [22]: class NQueens:

        def __init__(self, tamanho):
            self.tamanho = tamanho
            self.solucoes = 0
            self.fSolucao()

        def fSolucao(self):
            posicoes = [-1] * self.tamanho

```

```

self.insererRainha(posicoes, 0)
print("Show! Encontramos o total de:", self.solucoes, "soluções.")

def insererRainha(self, posicoes, alvoLinha):

    if alvoLinha == self.tamanho:
        self.exibeTabuleiro(posicoes)
        #self.exibeTabuleiroSimples(posicoes)
        self.solucoes += 1
    else:

        for coluna in range(self.tamanho):

            if self.checaPosicao(posicoes, alvoLinha, coluna):
                posicoes[alvoLinha] = coluna
                self.insererRainha(posicoes, alvoLinha + 1)
def checaPosicao(self, posicoes, posiOcupada, coluna):

    for i in range(posiOcupada):
        if posicoes[i] == coluna or \
            posicoes[i] - i == coluna - posiOcupada or \
            posicoes[i] + i == coluna + posiOcupada:
            return False
    return True

def exibeTabuleiro(self, posicoes):
    for tupla in range(self.tamanho):
        linha = ""
        for coluna in range(self.tamanho):
            if posicoes[tupla] == coluna:
                linha += "[R]"
            else:
                linha += "0"
        print(linha)
    print("\n")

def exibeTabuleiroSimples(self, posicoes):

    linha = ""
    for i in range(self.tamanho):
        linha += str(posicoes[i]) + " "
    print(linha)

def main():
    NQueens(8)
if __name__ == "__main__":
    # Definição de nome para rodar via script
    main()

```

```

[R]000000
0000[R]000
0000000[R]
00000[R]00
00[R]00000
000000[R]0
0[R]000000
000[R]0000

```

```

[R]0000000
00000[R]00
0000000[R]

```

00[R]00000
000000[R]0
000[R]0000
0[R]000000
0000[R]000

[R]0000000
000000[R]0
000[R]0000
00000[R]00
0000000[R]
0[R]000000
0000[R]000
00[R]00000

[R]0000000
000000[R]0
0000[R]000
0000000[R]
0[R]000000
000[R]0000
00000[R]00
00[R]00000

0[R]000000
000[R]0000
00000[R]00
0000000[R]
00[R]00000
[R]0000000
000000[R]0
0000[R]000

0[R]000000
0000[R]000
000000[R]0
[R]0000000
00[R]00000
0000000[R]
00000[R]00
000[R]0000

0[R]000000
0000[R]000
000000[R]0
000[R]0000
[R]0000000
0000000[R]
00000[R]00
00[R]00000

0[R]000000
00000[R]00
[R]0000000
000000[R]0
000[R]0000
0000000[R]
00[R]00000
0000[R]000

0[R]000000
00000[R]00
0000000[R]
00[R]00000
[R]0000000
000[R]0000
000000[R]0
0000[R]000

0[R]000000
000000[R]0
00[R]00000
00000[R]00
0000000[R]
0000[R]000
[R]0000000
000[R]0000

0[R]000000
000000[R]0
0000[R]000
0000000[R]
[R]0000000
000[R]0000
00000[R]00
00[R]00000

0[R]000000
0000000[R]
00000[R]00
[R]0000000
00[R]00000
0000[R]000
000000[R]0
000[R]0000

00[R]00000
[R]0000000
000000[R]0
0000[R]000
0000000[R]
0[R]000000
000[R]0000
00000[R]00

00[R]00000
0000[R]000
0[R]000000
0000000[R]
[R]0000000
000000[R]0
000[R]0000
00000[R]00

00[R]00000
0000[R]000
0[R]000000

0000000[R]
00000[R]00
000[R]0000
000000[R]0
[R]0000000

00[R]00000
0000[R]000
000000[R]0
[R]0000000
000[R]0000
0[R]000000
0000000[R]
00000[R]00

00[R]00000
0000[R]000
0000000[R]
000[R]0000
[R]0000000
000000[R]0
0[R]000000
00000[R]00

00[R]00000
00000[R]00
0[R]000000
0000[R]000
0000000[R]
[R]0000000
000000[R]0
000[R]0000

00[R]00000
00000[R]00
0[R]000000
000000[R]0
[R]0000000
000[R]0000
0000000[R]
0000[R]000

00[R]00000
00000[R]00
0[R]000000
000000[R]0
0000[R]000
[R]0000000
0000000[R]
000[R]0000

00[R]00000
00000[R]00
000[R]0000
[R]0000000
0000000[R]
0000[R]000
000000[R]0
0[R]000000

00[R]00000
00000[R]00
000[R]0000
0[R]000000
0000000[R]
0000[R]000
000000[R]0
[R]0000000

00[R]00000
00000[R]00
0000000[R]
[R]0000000
000[R]0000
000000[R]0
0000[R]000
0[R]000000

00[R]00000
00000[R]00
0000000[R]
[R]0000000
0000[R]000
000000[R]0
0[R]000000
000[R]0000

00[R]00000
00000[R]00
0000000[R]
0[R]000000
000[R]0000
[R]0000000
000000[R]0
0000[R]000

00[R]00000
000000[R]0
0[R]000000
0000000[R]
0000[R]000
[R]0000000
000[R]0000
00000[R]00

00[R]00000
000000[R]0
0[R]000000
0000000[R]
00000[R]00
000[R]0000
[R]0000000
0000[R]000

00[R]00000
0000000[R]
000[R]0000

000000[R]0
[R]0000000
00000[R]00
0[R]000000
0000[R]000

000[R]0000
[R]0000000
0000[R]000
0000000[R]
0[R]000000
000000[R]0
00[R]00000
00000[R]00

000[R]0000
[R]0000000
0000[R]000
0000000[R]
00000[R]00
00[R]00000
000000[R]0
0[R]000000

000[R]0000
0[R]000000
0000[R]000
0000000[R]
00000[R]00
[R]0000000
00[R]00000
000000[R]0

000[R]0000
0[R]000000
000000[R]0
00[R]00000
00000[R]00
0000000[R]
[R]0000000
0000[R]000

000[R]0000
0[R]000000
000000[R]0
00[R]00000
00000[R]00
0000000[R]
0000[R]000
[R]0000000

000[R]0000
0[R]000000
000000[R]0
0000[R]000
[R]0000000
0000000[R]
00000[R]00
00[R]00000

000[R]0000
0[R]000000
0000000[R]
0000[R]000
000000[R]0
[R]0000000
00[R]00000
00000[R]00

000[R]0000
0[R]000000
0000000[R]
00000[R]00
[R]0000000
00[R]00000
0000[R]000
000000[R]0

000[R]0000
00000[R]00
[R]0000000
0000[R]000
0[R]000000
0000000[R]
00[R]00000
000000[R]0

000[R]0000
00000[R]00
0000000[R]
0[R]000000
000000[R]0
[R]0000000
00[R]00000
0000[R]000

000[R]0000
00000[R]00
0000000[R]
00[R]00000
[R]0000000
000000[R]0
0000[R]000
0[R]000000

000[R]0000
000000[R]0
[R]0000000
0000000[R]
0000[R]000
0[R]000000
00000[R]00
00[R]00000

000[R]0000
000000[R]0
00[R]00000

0000000[R]
0[R]000000
0000[R]000
[R]0000000
00000[R]00

000[R]0000
000000[R]0
0000[R]000
0[R]000000
00000[R]00
[R]0000000
00[R]00000
0000000[R]

000[R]0000
000000[R]0
0000[R]000
00[R]00000
[R]0000000
00000[R]00
0000000[R]
0[R]000000

000[R]0000
0000000[R]
[R]0000000
00[R]00000
00000[R]00
0[R]000000
000000[R]0
0000[R]000

000[R]0000
0000000[R]
[R]0000000
0000[R]000
000000[R]0
0[R]000000
00000[R]00
00[R]00000

000[R]0000
0000000[R]
0000[R]000
00[R]00000
[R]0000000
000000[R]0
0[R]000000
00000[R]00

0000[R]000
[R]0000000
000[R]0000
00000[R]00
0000000[R]
0[R]000000
000000[R]0
00[R]00000

0000[R]000
[R]0000000
0000000[R]
000[R]0000
0[R]000000
000000[R]0
00[R]00000
00000[R]00

0000[R]000
[R]0000000
0000000[R]
00000[R]00
00[R]00000
000000[R]0
0[R]000000
000[R]0000

0000[R]000
0[R]000000
000[R]0000
00000[R]00
0000000[R]
00[R]00000
[R]0000000
000000[R]0

0000[R]000
0[R]000000
000[R]0000
000000[R]0
00[R]00000
0000000[R]
00000[R]00
[R]0000000

0000[R]000
0[R]000000
00000[R]00
[R]0000000
000000[R]0
000[R]0000
0000000[R]
00[R]00000

0000[R]000
0[R]000000
0000000[R]
[R]0000000
000[R]0000
000000[R]0
00[R]00000
00000[R]00

0000[R]000
00[R]00000
[R]0000000

00000[R]00
0000000[R]
0[R]000000
000[R]0000
000000[R]0

0000[R]000
00[R]00000
[R]0000000
000000[R]0
0[R]000000
0000000[R]
00000[R]00
000[R]0000

0000[R]000
00[R]00000
0000000[R]
000[R]0000
000000[R]0
[R]0000000
00000[R]00
0[R]000000

0000[R]000
000000[R]0
[R]0000000
00[R]00000
0000000[R]
00000[R]00
000[R]0000
0[R]000000

0000[R]000
000000[R]0
[R]0000000
000[R]0000
0[R]000000
0000000[R]
00000[R]00
00[R]00000

0000[R]000
000000[R]0
0[R]000000
000[R]0000
0000000[R]
[R]0000000
00[R]00000
00000[R]00

0000[R]000
000000[R]0
0[R]000000
00000[R]00
00[R]00000
[R]0000000
000[R]0000
0000000[R]

0000[R]000
000000[R]0
0[R]000000
00000[R]00
00[R]00000
[R]0000000
0000000[R]
000[R]0000

0000[R]000
000000[R]0
000[R]0000
[R]0000000
00[R]00000
0000000[R]
00000[R]00
0[R]000000

0000[R]000
0000000[R]
000[R]0000
[R]0000000
00[R]00000
00000[R]00
0[R]000000
000000[R]0

0000[R]000
0000000[R]
000[R]0000
[R]0000000
000000[R]0
0[R]000000
00000[R]00
00[R]00000

00000[R]00
[R]0000000
0000[R]000
0[R]000000
0000000[R]
00[R]00000
000000[R]0
000[R]0000

00000[R]00
0[R]000000
000000[R]0
[R]0000000
00[R]00000
0000[R]000
0000000[R]
000[R]0000

00000[R]00
0[R]000000
000000[R]0

[R]0000000
000[R]0000
0000000[R]
0000[R]000
00[R]00000

00000[R]00
00[R]00000
[R]0000000
000000[R]0
0000[R]000
0000000[R]
0[R]000000
000[R]0000

00000[R]00
00[R]00000
[R]0000000
0000000[R]
000[R]0000
0[R]000000
000000[R]0
0000[R]000

00000[R]00
00[R]00000
[R]0000000
0000000[R]
0000[R]000
0[R]000000
000[R]0000
000000[R]0

00000[R]00
00[R]00000
0000[R]000
000000[R]0
[R]0000000
000[R]0000
0[R]000000
0000000[R]

00000[R]00
00[R]00000
0000[R]000
0000000[R]
[R]0000000
000[R]0000
0[R]000000
000000[R]0

00000[R]00
00[R]00000
000000[R]0
0[R]000000
000[R]0000
0000000[R]
[R]0000000
0000[R]000

00000[R]00
00[R]00000
000000[R]0
0[R]000000
0000000[R]
0000[R]000
[R]0000000
000[R]0000

00000[R]00
00[R]00000
000000[R]0
000[R]0000
[R]0000000
0000000[R]
0[R]000000
0000[R]000

00000[R]00
000[R]0000
[R]0000000
0000[R]000
0000000[R]
0[R]000000
000000[R]0
00[R]00000

00000[R]00
000[R]0000
0[R]000000
0000000[R]
0000[R]000
000000[R]0
[R]0000000
00[R]00000

00000[R]00
000[R]0000
000000[R]0
[R]0000000
00[R]00000
0000[R]000
0[R]000000
0000000[R]

00000[R]00
000[R]0000
000000[R]0
[R]0000000
0000000[R]
0[R]000000
0000[R]000
00[R]00000

00000[R]00
0000000[R]
0[R]000000

000[R]0000
[R]0000000
000000[R]0
0000[R]000
00[R]00000

000000[R]0
[R]0000000
00[R]00000
0000000[R]
00000[R]00
000[R]0000
0[R]000000
0000[R]000

000000[R]0
0[R]000000
000[R]0000
[R]0000000
0000000[R]
0000[R]000
00[R]00000
00000[R]00

000000[R]0
0[R]000000
00000[R]00
00[R]00000
[R]0000000
000[R]0000
0000000[R]
0000[R]000

000000[R]0
00[R]00000
[R]0000000
00000[R]00
0000000[R]
0000[R]000
0[R]000000
000[R]0000

000000[R]0
00[R]00000
0000000[R]
0[R]000000
0000[R]000
[R]0000000
00000[R]00
000[R]0000

000000[R]0
000[R]0000
0[R]000000
0000[R]000
0000000[R]
[R]0000000
00[R]00000
00000[R]00

000000[R]0
000[R]0000
0[R]000000
0000000[R]
00000[R]00
[R]0000000
00[R]00000
0000[R]000

000000[R]0
0000[R]000
00[R]00000
[R]0000000
00000[R]00
0000000[R]
0[R]000000
000[R]0000

0000000[R]
0[R]000000
000[R]0000
[R]0000000
000000[R]0
0000[R]000
00[R]00000
00000[R]00

0000000[R]
0[R]000000
0000[R]000
00[R]00000
[R]0000000
000000[R]0
000[R]0000
00000[R]00

0000000[R]
00[R]00000
[R]0000000
00000[R]00
0[R]000000
0000[R]000
000000[R]0
000[R]0000

0000000[R]
000[R]0000
[R]0000000
00[R]00000
00000[R]00
0[R]000000
000000[R]0
0000[R]000

Show! Encontramos o total de: 92 soluções.

In []:

