

مسئله n وزیر یک مسئله کلاسیک در تئوری صفحه‌بندی (Chessboard) است که هدف آن قرار دادن n عدد وزیر بر روی صفحه شطرنج به گونه‌ای است که هیچکدام از وزیرها به یکدیگر حمله نکنند. در زبان پایتون، می‌توانیم این مسئله را به صورت زیر حل کنیم:

```
```python
```

```
def is_safe(board, row, col, n):
```

```
    for i in range(row):
```

```
        if board[i] == col or \
```

```
            board[i] - i == col - row or \
```

```
            board[i] + i == col + row:
```

```
        return False
```

```
    return True
```

```
def solve_n_queens(n, row, board, result):
```

```
    if row == n:
```

```
        result.append(board[:])
```

```
        return
```

```
    for col in range(n):
```

```
        if is_safe(board, row, col, n):
```

```
            board[row] = col
```

```
            solve_n_queens(n, row + 1, board, result)
```

```
def n_queens(n):
```

```
    result = []
```

```
    solve_n_queens(n, 0, [-1] * n, result)
```

```
    return result
```

```
n = 8 # تعداد وزیرها
```

```
solutions = n_queens(n)
```

```
for solution in solutions:
```

```
    print(solution)
```

```
'''
```

توضیح خط به خط:

1. ``is_safe``: یک تابع که بررسی می‌کند آیا می‌توان یک وزیر را در یک موقعیت خاص قرار داد یا خیر.

2. ``solve_n_queens``: تابع بازگشتی برای حل مسئله  $n$  وزیر.

3. ``n_queens``: تابع اصلی که تعداد را ورودی گرفته و تمامی حالت‌های ممکن را برای قرار دادن وزیرها بر روی صفحه شطرنج برمی‌گرداند.

4. در نهایت، ما تعداد حالات مختلف برای قرار دادن  $n$  وزیر بر روی صفحه شطرنج را چاپ می‌کنیم.