[CS 11] Prac 8k – King Threats

Problem Statement

Consider an $r \times c$ chessboard with some chess pieces in it. For simplicity, we ignore the actual rules of chess. For example, chess pieces don't have "color".

We say a square on the chessboard is **unsafe** if a piece can reach it in one move.

In one move, a king can travel *one* square horizontally, vertically, or diagonally.

Given the locations of kings, label which squares are unsafe.

Task Details

Your task is to implement a function called `king_threats`. This function has a single parameter: a `tuple` of r `str`s, each of which is c characters long and represents a row. Each character represents a square and is:

- a `.` if it is free;
- a `#` if it contains a king.

The function must return a `list` of `str`s representing the same chessboard, but with the unsafe squares labelled with `*`.

Restrictions

(See 8a for more restrictions)

For this problem:

- Loops and lists are allowed.
- Up to 18 function definitions are allowed.
- Recursion is **disallowed**. (The recursion limit has been greatly reduced.)
- Sets and dictionaries are allowed.
- Generators and comprehensions are allowed.
- The source code limit is 6000.

Example Calls

Example 1 Function Call

```
king_threats((
    '.............',
    '..........#..',
    '.............',
    '...#.........',
    '..#.........#',
    '.............',
    '.............',
    '.............',
))
```

Copy

Example 1 Return Value

```
[
    '.........***.',
    '.........*#*.',
    '..***....***.',
    '.**#*......**',
    '.*#**......*#',
    '.***.......**',
    '.............',
    '.............',
]
```

Copy

Example 1 Explanation

Hint: You can print a grid of `str`s by doing:

```
for row in grid:
    print(row)
```

Copy

or by doing

```
print(*grid, sep='\n')
```

Copy

Constraints

- The function `king_threats` will be called at most $1{,}000$ times.
- The sum of the rc across all calls will be at most $500{,}000$.
- $1 \le r \le 100$
- $1 \le c \le 5{,}000$

Scoring

- You get 100 ♥ points if you solve all test cases where:
  - $r, c \le 50$
  - the sum of the rc across all calls will be at most $10{,}000$.
- You get 30 ♥ points if you solve all test cases where:
  - there are at most 5000 pieces.
- You get 30 ♥ points if you solve all test cases.

Clarifications

Report an issue

No clarifications have been made at this time.

Sub mit solution [CS 11] Practice 8 ♥

♥

✔ **Points:** 160 (partial)

Time limit: 6.0s

Memory limit: 1G

Author: kvatienza (Kevin Atienza)

› **Problem type**

⌄ **Allowed languages**
~~NONE~~, py3