Problem 1222: Queens That Can Attack the King

Problem Information

Difficulty: Medium
Acceptance Rate: 72.47%
Paid Only: No
Tags: Array, Matrix, Simulation

Problem Description

On a **0-indexed** `8 x 8` chessboard, there can be multiple black queens and one white king.

You are given a 2D integer array `queens` where `queens[i] = [xQueeni, yQueeni]` represents the position of the `ith` black queen on the chessboard. You are also given an integer array `king` of length `2` where `king = [xKing, yKing]` represents the position of the white king.

Return _the coordinates of the black queens that can directly attack the king_. You may return the answer in **any order**.

Example 1:

Input: queens = [[0,1],[1,0],[4,0],[0,4],[3,3],[2,4]], king = [0,0] **Output:** [[0,1],[1,0],[3,3]] **Explanation:** The diagram above shows the three queens that can directly attack the king and the three queens that cannot attack the king (i.e., marked with red dashes).

Example 2:

Input: queens = [[0,0],[1,1],[2,2],[3,4],[3,5],[4,4],[4,5]], king = [3,3] **Output:** [[2,2],[3,4],[4,4]] **Explanation:** The diagram above shows the three queens that can directly attack the king and the three queens that cannot attack the king (i.e., marked with red

dashes).

Constraints:

* `1 <= queens.length < 64` * `queens[i].length == king.length == 2` * `0 <= xQueeni, yQueeni, xKing, yKing < 8` * All the given positions are **unique**.

Code Snippets

C++:

```
class Solution {
public:
vector<vector<int>> queensAttacktheKing(vector<vector<int>>& queens,
vector<int>& king) {

}
};
```

Java:

```
class Solution {
public List<List<Integer>> queensAttacktheKing(int[][] queens, int[] king) {

}
}
```

Python3:

```
class Solution:
def queensAttacktheKing(self, queens: List[List[int]], king: List[int]) ->
List[List[int]]:
```