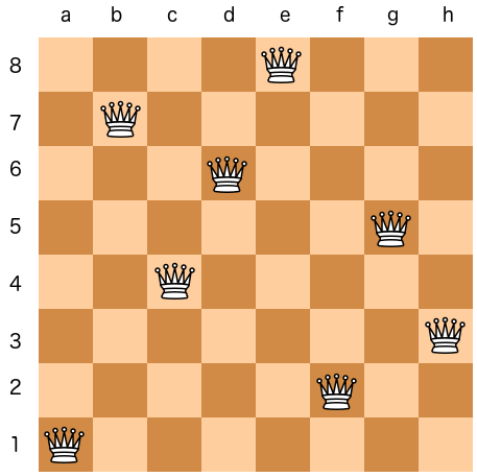
EE2310 Introduction to Programming – Assignment 3
Due: Jan 15th, 2021 (11:59pm)

I. Submission

1. You must submit your source file containing the designated function at NTHU OJ by the deadline (like you did in the Lab classes).
2. You must prepare a **README file** in **.txt** format (~300 words, 2~3 paragraphs) that summarizes how you write your program, e.g. what your variables or functions are for, etc.
3. Prepare your source file (of the designated function) in a single **.c** file. No need to submit the main function.
4. Compress your README and source files in a single zipped file named `YourStudentID.zip` and submit this zipped file onto the **eLearn system**.
5. Late submission will incur 10% penalty per day up to 3 days. After that, assignment submission will be closed and no submissions will be accepted.

II. Task Description

Write a program that finds all possible ways to place K queens on an K×K chessboard so that no two queens threaten each other. Thus, a solution requires that no two queens share the same row, column, or diagonal, as shown in the diagram below:



Let (1,1) be the top-left corner and x, y be the vertical as well as horizontal axes, respectively. We represent this solution as (1,5),(2,2),(3,4),(4,7),(5,3),(6,8),(7,6),(8,1).
For simplicity, since the x-axis is counted according to numerical order, we can simply represent this solution as the following 8 numbers: 5 2 4 7 3 8 6 1.

III. What you need to do?

We will provide the main function that deals with I/O. The input will be the number of K. All you need to do is to output ALL POSSIBLE SOLUTIONS. If we count different rotations and flippings as different solutions, then there are 92 solutions for case K=8.
You should output these solutions in **lexicographic order** (e.g. 15863724 should be placed before 17468253). Your program should have reasonable efficiency and you may need to use a stack to do that.

IV. Sample I/O

- **Sample input (Case 1)**

```
8
```

- **Sample output (Case 1)**

```
1 5 8 6 3 7 2 4
1 6 8 3 7 4 2 5
1 7 4 6 8 2 5 3
1 7 5 8 2 4 6 3
        .
        .
        .
8 2 4 1 7 5 3 6
8 2 5 3 1 7 4 6
8 3 1 6 2 5 7 4
8 4 1 3 6 2 7 5
A total of 92 solutions for 8-Queens problem.
```

- **Sample input (Case 2)**

```
2
```

- **Sample output (Case 2)**

```
A total of 0 solutions for 2-Queens problem.
```

V. Guidelines

1. Assessment: Correctness 70%, Program Style 30%.

2. Correctness:

   Your program output will be checked automatically by NTHU OJ. This time we already took care of the I/O and there should be no presentation errors anymore. Just save your result in the designated array.

3. Program Style:

   1. Your programs should be *well-commented*. All variables, functions, loops should be clearly explained both in the source code and briefly in the README file.
   2. Blocks in your source code should have *proper indentation* (縮排).
   3. Do not use any global variable to pass data in and out of a function.
   4. For a good program style, please refer to the Linux Kernel Coding Style at https://www.kernel.org/doc/html/v4.10/process/coding-style.html

This is not an easy assignment.
Please start early!!