

CS 2123 Data Structures
Recitation 2
Due Friday February 12

Difficulty *** (out of 5)

1. (100 pts) Write a program to solve the *NQueens* problem recursively. *NQueens* problem investigates whether it is possible to place N queens on an $N \times N$ chessboard so that none of them can move to a square occupied by any of the others in a single turn. Your program should either display a solution if it finds one or report that no solution exists.

A single queen and squares that can be occupied in a single turn are shown below.

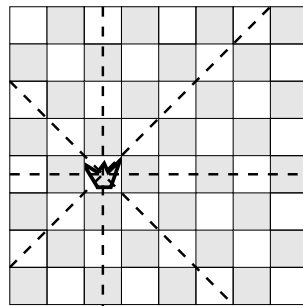


Figure 1: Single Queen

A sample solution to the 8-queens problem is given below.

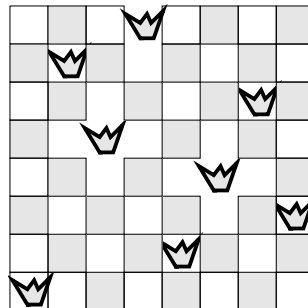


Figure 2: Solution for 8-queens problem

Below are some hints to work on the problem.

- Use a dynamically allocated 2D array to store the positions of queens on the board.
- Number the rows and columns from zero.
- Note that only one Queen can occupy each column.
- Each column **must** have a Queen.
- Move across the grid, column by column. Place a queen in each column. Start with column 0 and move towards column $N-1$.

- Place the queen for the current column in a row and diagonal such that she does not threaten previously placed queens.

Sample Execution of the program is given below

```
Enter n
8
Solution:
X_____
_____X_
____X____
_____X
_X______
___X_____
____X____
__X_____
```

Another sample execution is given below

```
Enter n
3
NO SOLUTION
```

- You can work in groups of 2-3
- You can work on this during two weeks of recitations.
- Dynamically allocate the array after reading the dimensions from the file.

Test your program with different values of N . It is easy to see whether a placement is valid or not. You can write a function to verify a solution to test your program if you want.

Submit your program electronically using the blackboard system. Only one member of the group should send it. List the name of the group members on the top of your submission.