#include <stdio.h>

#define MAX 100 // define maximum board size

int board[MAX]; // array to store column positions

int count\_Ts = 0; //for count total solution

//check row,col for placing a queen in a safe position

int checkissafe(int n, int r, int c)

{

for (int i = 0; i < r; i++)

{

int j = board[i];

if (j == c) // Check for column

{

return 0;

}

if ((r-i) == (c-j) || (r-i) == (j-c)) // Check for diagonal

{

return 0;

}

}

return 1;

}

void NQueens(int r, int n)

{

if (r == n) // all queens are placed

{

count\_Ts++; //increment total counter

printf("Solution %d:\n", count\_Ts);

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

{

if (board[i] == j) // if queen is placed then print Q otherwise #

{

printf("Q");

}

else

{

printf("#");

}

}

printf("\n");

}

printf("\n");

return;

}

// For placing a queen in each column of the current row

for (int c = 0; c < n; c++)

{

if (checkissafe(n,r,c))

{

board[r] = c; // Place the queen

NQueens(r + 1, n); // Recursively call the next row

}

}

}

int main()

{

int n;

printf("Enter the value N: ");

scanf("%d",&n);

if (n > MAX) //for checking maxium number

{

printf("Maximum value of N: %d\n", MAX);

return 0;

}

NQueens(0,n); //call solve function, start row from 0

printf("Total Solutions: %d\n", count\_Ts); //finally print total solution

return 0;

}