NQueens

:-1BM21CS232

```
#include <stdio.h>

int n, count=0;

int isSafe(char board[n][n], int row, int col)
{
        int i,j;
   for (i= row - 1; i >= 0; i--)
   {
      if (board[i][col] == 'Q')
      {
         return 0;
      }
   }

   for (i = row - 1, j = col - 1; i >= 0 && j >= 0; i--, j--)
   {
      if (board[i][j] == 'Q')
      {
         return 0;
      }
   }

   for (i = row - 1, j = col + 1; i >= 0 && j < n; i--, j++)
   {
      if (board[i][j] == 'Q')
      {
         return 0;
      }
   }
   return 1;
}
```

```
void printBoard(char board[][n])
{
   printf("\n---Chess Board---\n");
        int i,j;
   for (i = 0; i < n; i++)
   {
      for (j = 0; j < n; j++)
      {
         printf("%c ", board[i][j]);
      }
      printf("\n");
   }
}

void nQueens(char board[n][n], int row)
{
   if (row == n)
   {
      printBoard(board);
      count++;
      return;
   }
      int j;
   for ( j= 0; j < n; j++)
   {
      if (isSafe(board, row, j) == 1)
      {
         board[row][j] = 'Q';
         nQueens(board, row + 1);
         board[row][j] = 'X';
      }
   }
}
```

```
int main()
{
   printf("Enter the size of the board: ");
   scanf("%d", &n);

   char board[n][n];
        int i,j;
   for ( i= 0; i < n; i++)
   {
      for (j = 0; j < n; j++)
      {
         board[i][j] = 'X';
      }
   }
   nQueens(board, 0);
   printf("\nTotal Possible Solution: %d ",count);
}
```

OUTPUT: