

ASSIGNMENT #6 – N Queens Problem

C Code of 8 Queens Problem

```
#include <stdio.h>
#define QUEENS 8          /* number of queens and board size */
#define int unsigned

int number = 0;
int board [QUEENS];      /* each element represents a queue*/

int boardTamam            /* check the validity of the board */
    (register int queue)
{
    register int r;

    for (r = 0; r < queue; r++) /* Check all previous queues */
        if (board[queue] == board[r] ||
            board[queue] == board[r] << queue-r ||
            board[queue] == board[r] >> queue-r)
            return 0;
    return 1;
}

void writesol (void)      /* show the solution; Increase the NUMBER */
{
    register int t, r;

    printf("\n\n\t SOLUTIONS %u\n\n", ++number);
    for (r = 0; r < QUEENS; r++) { /* queue */
        for (t = 1<<QUEENS-1; t > 0; t >>= 1)
            printf(" %c", board[r] == t ? 'Q' : '.');
        printf("\n");
    }
}

void place (int queue)    /* place to next queue */
{
    if (queue == QUEENS)  /* all queues are full and checked */
        writesol();
    else
        for (board[queue]=1; board[queue]<1<<QUEENS; board[queue]<<=1)
            if (boardTamam(queue))
                place(queue+1); /* try the next queue */
}

signed main (void)
{
    place(0);
    printf("\n\n There is %u diffirent solution for Queen Promlem \n",
QUEENS, number);
    return 0;
}
```

Output of Program

8queens

SOLUTIONS 1

```
. . . . . Q
. . . Q . . .
Q . . . . .
. . Q . . . .
. . . . Q . .
. Q . . . . .
. . . . . Q .
. . . . Q . .
```

SOLUTIONS 2

```
. . . . . Q
. . Q . . . .
Q . . . . .
. . . . Q . .
. Q . . . . .
. . . Q . . .
. . . . . Q .
. . . Q . . .
```

SOLUTIONS 3

```
. . . . . Q
. Q . . . . .
. . . . Q . .
. . Q . . . .
Q . . . . .
. . . . . Q .
. . . Q . . .
. . . . . Q .
```

SOLUTIONS 4

```
. . . . . Q
. Q . . . . .
. . . Q . . .
Q . . . . .
. . . . . Q .
. . . Q . . .
. . Q . . . .
. . . . . Q .
```

SOLUTIONS 46

```
. . . . Q . .
Q . . . . .
. . . Q . . .
. . . . . Q .
. Q . . . . Q
. . . . . Q .
. . . . . Q .
. . Q . . . .
```

SOLUTIONS 47

```
. . . Q . . .
. . . . . Q
. . . . Q . .
. . Q . . . .
Q . . . . .
. . . . . Q
. Q . . . . .
. . . . . Q .
```

SOLUTIONS 48

```
. . . Q . . .
. . . . . Q
Q . . . . .
. . . . Q . .
. . . . . Q
. Q . . . . .
. . . . . Q
. . Q . . . .
```

SOLUTIONS 49

```
. . . Q . . .
. . . . . Q
Q . . . . .
. . Q . . . .
. Q . . . . .
. . . . . Q
. . . . . Q
. . . . Q . .
```

SOLUTIONS 89

```
Q . . . . .
. . . . . Q
. . . . Q . .
. . . . . Q
. Q . . . . .
. . . Q . . .
. . . . . Q
. . Q . . . .
```

SOLUTIONS 90

```
Q . . . . .
. . . . . Q
. . . . Q . .
. . . . . Q
. . . . . Q
. Q . . . . .
. . . Q . . .
. . Q . . . .
```

SOLUTIONS 91

```
Q . . . . .
. . . . . Q
. . . . . Q
. . Q . . . .
. . . . . Q
. . . Q . . .
. Q . . . . .
. . . Q . . .
```

SOLUTIONS 92

```
Q . . . . .
. . . . . Q
. . . . . Q
. . . . . Q
. . Q . . . .
. . . . . Q
. Q . . . . .
. . . Q . . .
```

There is 92 diffirent solution for Queen Promlem

Queen Promlem's took 0.757000 seconds to execute

The solution algorithm of this problem is LINEAR.

Complexity of the program : $O(n)$

[Download the C Code File \(until 24 January.\)](#)