In this session we revisit the Queens Problem: given \( n \geq 4 \), place \( n \) queens on an \( n \times n \) chess board, so that they do not attack each other (a queen attacks pieces on the same row, on the same column, and on the same diagonals).

Write a program `queens` with CPLEX that, given \( n \) from the command line, outputs a solution as follows:

```
$ queens 8
.......X
..X....
X.......X
..X.....
......X.
.X......X
.....X.
.X.......
```

Which is the least \( n \) for which your program takes more than 5 secs. in your computer? How long does it take to solve the queens problem for this \( n \)?

Use the `run` script below, which is also available at the website of CPS (http://www.cs.upc.edu/~erodri/cps.html):

```bash
#!/bin/bash

for i in $(seq 4 1000); do
    echo "i:$i"
    time ./$1 $i
done
```

Compare the performance with that of the programs implemented in the previous session on Queens Problem.