# CS5384 Logic for Computer Scientists
# Satisfiability test of clauses and its application in n-queens problem.

**Ayyappa Reddy Maramreddy (R11888944)**

**Sanjana Chitteni (R11844919)**

**Problem Description:**

The objective of the problem statement is to find a formula whose satisfiable assignment gives the solution to the n-queens problem. Take a nxn chess board in which each position on a chess board is denoted by a propositional letter Pij for position (i, j). Place n-queens on a nxn chess board such that no two queens should be in the same column, no two queens on the same row and no two queens on the same diagonal.

**Problem solution:**

In the problem solution the proposition logic is represented in the conjunctive normal form (CNF). The logic designed will generate a CNF formula based on the size of the chess board, which is nothing but the input to the sat solver.

**SAT Solver:**

In computer science, a sat solver is a program which helps in solving Boolean satisfiability by taking a CNF formula as an input and outputs whether the formula is satisfiable.

**Conjunctive normal form (CNF):**

A formula in Boolean logic is said to be in conjunctive normal form (CNF) if it is a conjunction of at least one clause (disjunction of literals).

**Encoding the problem into CNF form:**

Assume that Pij is a propositional letter for position (i, j) in the nxn chess board. According to the problem statement of n-queens problem there should be only one queen in each row, column, and diagonal. For example, let's consider 7 queens' problem, to have a queen in the first row there are 7 possibilities mentioned below.

(P11 & ~P12 & ~P13 & ~P14 & ~P15 & ~ P16 & ~ P17 ) V (~ P11 & P12 & ~P13 & ~P14& ~P15& ~ P16 & ~ P17 ) V (~ P11 & ~P12 & P13 & ~P14& ~P15 &~ P16 & ~ P17) V (~ P11 & ~P12 & ~P13 & P14& ~P15& ~ P16 & ~ P17) V (~ P11 & ~P12 & ~P13 & ~ P14& P15& ~ P16 & ~ P17) V (~ P11 & ~P12 & ~P13 & ~P14& ~P15 & P16 & ~ P17) V (~ P11 & ~P12 & ~P13 & ~P14& ~P15 & P16 & ~ P17).

The above format is not in the form of CNF format. Let convert it into conjunction normal form.

Let's consider that there is a queen in position (1,1) then there should not be any queen on any other position of that row. Symbolically it can be written as

P11 V P12 V P13 V P14 V P14 V P15 V P16 V P17 (exactly one of the positions will be true)

*P11 → ~ P12 => ~ P11 V ~ P12*

*P11 → ~ P13 => ~ P11 V ~ P13*

*P11 → ~ P14 => ~ P11 V ~ P14*

*P11 → ~ P15 => ~ P11 V ~ P15*

*P11 → ~ P16 => ~ P11 V ~ P16*

*P11 → ~ P17 => ~ P11 V ~ P17*

Out of the above 7 clauses, if we take first clause in conjunction with any other clause, it results in exactly one position Pij as true. The same rules will be applicable for other rows and columns.

**<u>Pseudocode for Generating CNF File</u>**:

```
Begin
if n<4
    print "Unsatisfiable"
    return
else
    for each row
            generate clause with all the proposition letters in the row
            generate clauses (2 negative propositions with each clause) with all combinations
            in a row
    for each column
            generate clause with all the proposition letters in a column
            generate clauses (2 negative propositions in each clause) with all combinations
            in column
    for all diagonals in the lower triangle from left to right
            take 2 propositions (negative) at a time and generate all the possible clauses.
    for all diagonals in the upper triangle from left to right
            take 2 propositions (negative) at a time and generate all the possible clauses.
    for all diagonals in the bottom triangle from right to left
            take 2 propositions (negative) at a time and generate all the possible clauses.
    for all diagonals in the upper triangle from right to left
            take 2 propositions (negative) at a time and generate all the possible clauses.
```

*Save all the generated clauses into a file with CNF format.*

*end*

**Executing the generated CNF file**

Next step is to execute the above generated CNF formula in a miniSAT solver. The miniSAT solver takes the above generated CNF formula as an input and tells us whether the formula is satisfiable or not. If satisfiable, it gives the proposition sequence with positive and negative values where a positive proposition indicates there is a queen in that position and vice versa.

## 3 - Queens Problem:

- The 3 queens' problem is unsatisfiable because it is not satisfying the conditions mentioned above.
- In a 3*3 chess boards there will be 3 queens.
- When we are trying to arrange the queens in an order that satisfies all conditions.
- Queens in the board are either horizontally or vertically or diagonally connecting to each other.
- So, the N-queens problem statement is unsatisfiable.

**Sample output for 3 queens**

```
Please enter n-queen value:
3
N-queens problem for the given N is not satisfiable. Please enter a value greater than 3
Please enter n-queen value:
```

## 7 - Queens Problem:

- The 7 queens' problem is satisfiable because it is satisfying the conditions mentioned above.
- In a 7*7 chess boards there will be 7 queens.
- When we are trying to arrange the queens in an order that satisfies all conditions.
- queen can move or attack in vertical, horizontal, diagonal ways on the board and no two queens in the board are attacking each other.
- So, the N-queens problem statement is satisfiable.

**Sample output for 7 queens:**

**c There are 49 positions**

**c the clauses for 7-queen are below**

**p cnf 49 490**

**1 2 3 4 5 6 7 0**

**-1 -2 0**

**-1 -3 0**

**-1 -4 0**

**-1 -5 0**

**-1 -6 0**

**-1 -7 0**

**-2 -3 0**

**-2 -4 0**

**-2 -5 0**

**-2 -6 0**

**-2 -7 0**

**-3 -4 0**

**-3 -5 0**

**-3 -6 0**

**-3 -7 0**

**-4 -5 0**

**-4 -6 0**

**-4 -7 0**

**-5 -6 0**

**-5 -7 0**

**-6 -7 0**

**8 9 10 11 12 13 14 0**

**-8 -9 0**

**-8 -10 0**

**-8 -11 0**

**-8 -12 0**

**-8 -13 0**

**-8 -14 0**

**-9 -10 0**

**-9 -11 0**

**-9 -12 0**

**-9 -13 0**

**-9 -14 0**

**-10 -11 0**

**-10 -12 0**

**-10 -13 0**

**-10 -14 0**

**-11 -12 0**

**-11 -13 0**

**-11 -14 0**

**-12 -13 0**

**-12 -14 0**

**-13 -14 0**

**15 16 17 18 19 20 21 0**

**-15 -16 0**

**-15 -17 0**

**-15 -18 0**

**-15 -19 0**

**-15 -20 0**

**-15 -21 0**

**-16 -17 0**

**-16 -18 0**

**-16 -19 0**

**-16 -20 0**

**-16 -21 0**

**-17 -18 0**

**-17 -19 0**

**-17 -20 0**

**-17 -21 0**

**-18 -19 0**

**-18 -20 0**

**-18 -21 0**

**-19 -20 0**

**-19 -21 0**

**-20 -21 0**

**22 23 24 25 26 27 28 0**

**-22 -23 0**

**-22 -24 0**

**-22 -25 0**

**-22 -26 0**

**-22 -27 0**

**-22 -28 0**

**-23 -24 0**

**-23 -25 0**

**-23 -26 0**

**-23 -27 0**

**-23 -28 0**

**-24 -25 0**

**-24 -26 0**

**-24 -27 0**

**-24 -28 0**

**-25 -26 0**

**-25 -27 0**

**-25 -28 0**

**-26 -27 0**

**-26 -28 0**

**-27 -28 0**

**29 30 31 32 33 34 35 0**

**-29 -30 0**

**-29 -31 0**

**-29 -32 0**

**-29 -33 0**

**-29 -34 0**

**-29 -35 0**

**-30 -31 0**

**-30 -32 0**

**-30 -33 0**

**-30 -34 0**

**-30 -35 0**

**-31 -32 0**

**-31 -33 0**

**-31 -34 0**

**-31 -35 0**

**-32 -33 0**

**-32 -34 0**

**-32 -35 0**

**-33 -34 0**

**-33 -35 0**

**-34 -35 0**

**36 37 38 39 40 41 42 0**

**-36 -37 0**

**-36 -38 0**

**-36 -39 0**

**-36 -40 0**

**-36 -41 0**

**-36 -42 0**

**-37 -38 0**

**-37 -39 0**

**-37 -40 0**

**-37 -41 0**

**-37 -42 0**

**-38 -39 0**

**-38 -40 0**

**-38 -41 0**

**-38 -42 0**

**-39 -40 0**

**-39 -41 0**

**-39 -42 0**

**-40 -41 0**

**-40 -42 0**

**-41 -42 0**

**43 44 45 46 47 48 49 0**

**-43 -44 0**

**-43 -45 0**

**-43 -46 0**

**-43 -47 0**

**-43 -48 0**

**-43 -49 0**

**-44 -45 0**

**-44 -46 0**

**-44 -47 0**

**-44 -48 0**

**-44 -49 0**

**-45 -46 0**

**-45 -47 0**

**-45 -48 0**

**-45 -49 0**

**-46 -47 0**

**-46 -48 0**

**-46 -49 0**

**-47 -48 0**

**-47 -49 0**

**-48 -49 0**

**1 8 15 22 29 36 43 0**

**-1 -8 0**

**-1 -15 0**

**-1 -22 0**

**-1 -29 0**

**-1 -36 0**

**-1 -43 0**

**-8 -15 0**

**-8 -22 0**

**-8 -29 0**

**-8 -36 0**

**-8 -43 0**

**-15 -22 0**

**-15 -29 0**

**-15 -36 0**

**-15 -43 0**

**-22 -29 0**

**-22 -36 0**

**-22 -43 0**

**-29 -36 0**

**-29 -43 0**

**-36 -43 0**

**2 9 16 23 30 37 44 0**

**-2 -9 0**

**-2 -16 0**

**-2 -23 0**

**-2 -30 0**

**-2 -37 0**

**-2 -44 0**

**-9 -16 0**

**-9 -23 0**

**-9 -30 0**

**-9 -37 0**

**-9 -44 0**

**-16 -23 0**

**-16 -30 0**

**-16 -37 0**

**-16 -44 0**

**-23 -30 0**

**-23 -37 0**

**-23 -44 0**

**-30 -37 0**

**-30 -44 0**

**-37 -44 0**

**3 10 17 24 31 38 45 0**

**-3 -10 0**

**-3 -17 0**

**-3 -24 0**

**-3 -31 0**

**-3 -38 0**

**-3 -45 0**

**-10 -17 0**

**-10 -24 0**

**-10 -31 0**

**-10 -38 0**

**-10 -45 0**

**-17 -24 0**

**-17 -31 0**

**-17 -38 0**

**-17 -45 0**

**-24 -31 0**

**-24 -38 0**

**-24 -45 0**

**-31 -38 0**

**-31 -45 0**

**-38 -45 0**

**4 11 18 25 32 39 46 0**

**-4 -11 0**

**-4 -18 0**

**-4 -25 0**

**-4 -32 0**

**-4 -39 0**

**-4 -46 0**

**-11 -18 0**

**-11 -25 0**

**-11 -32 0**

**-11 -39 0**

**-11 -46 0**

**-18 -25 0**

**-18 -32 0**

**-18 -39 0**

**-18 -46 0**

**-25 -32 0**

**-25 -39 0**

**-25 -46 0**

**-32 -39 0**

**-32 -46 0**

**-39 -46 0**

**5 12 19 26 33 40 47 0**

**-5 -12 0**

**-5 -19 0**

**-5 -26 0**

**-5 -33 0**

**-5 -40 0**

**-5 -47 0**

**-12 -19 0**

**-12 -26 0**

**-12 -33 0**

**-12 -40 0**

**-12 -47 0**

**-19 -26 0**

**-19 -33 0**

**-19 -40 0**

**-19 -47 0**

**-26 -33 0**

**-26 -40 0**

**-26 -47 0**

**-33 -40 0**

**-33 -47 0**

**-40 -47 0**

**6 13 20 27 34 41 48 0**

**-6 -13 0**

**-6 -20 0**

**-6 -27 0**

**-6 -34 0**

**-6 -41 0**

**-6 -48 0**

**-13 -20 0**

**-13 -27 0**

**-13 -34 0**

**-13 -41 0**

**-13 -48 0**

**-20 -27 0**

**-20 -34 0**

**-20 -41 0**

**-20 -48 0**

**-27 -34 0**

**-27 -41 0**

**-27 -48 0**

**-34 -41 0**

**-34 -48 0**

**-41 -48 0**

**7 14 21 28 35 42 49 0**

**-7 -14 0**

**-7 -21 0**

**-7 -28 0**

**-7 -35 0**

**-7 -42 0**

**-7 -49 0**

**-14 -21 0**

**-14 -28 0**

**-14 -35 0**

**-14 -42 0**

**-14 -49 0**

**-21 -28 0**

**-21 -35 0**

**-21 -42 0**

**-21 -49 0**

**-28 -35 0**

**-28 -42 0**

**-28 -49 0**

**-35 -42 0**

**-35 -49 0**

**-42 -49 0**

**-36 -44 0**

**-29 -37 0**

**-29 -45 0**

**-37 -45 0**

**-22 -30 0**

**-22 -38 0**

**-22 -46 0**

**-30 -38 0**

**-30 -46 0**

**-38 -46 0**

**-15 -23 0**

**-15 -31 0**

**-15 -39 0**

**-15 -47 0**

**-23 -31 0**

**-23 -39 0**

**-23 -47 0**

**-31 -39 0**

**-31 -47 0**

**-39 -47 0**

**-8 -16 0**

**-8 -24 0**

**-8 -32 0**

**-8 -40 0**

**-8 -48 0**

**-16 -24 0**

**-16 -32 0**

**-16 -40 0**

**-16 -48 0**

**-24 -32 0**

**-24 -40 0**

**-24 -48 0**

**-32 -40 0**

**-32 -48 0**

**-40 -48 0**

**-1 -9 0**

**-1 -17 0**

**-1 -25 0**

**-1 -33 0**

**-1 -41 0**

**-1 -49 0**

**-9 -17 0**

**-9 -25 0**

**-9 -33 0**

**-9 -41 0**

**-9 -49 0**

**-17 -25 0**

**-17 -33 0**

**-17 -41 0**

**-17 -49 0**

**-25 -33 0**

**-25 -41 0**

**-25 -49 0**

**-33 -41 0**

**-33 -49 0**

**-41 -49 0**

**-2 -10 0**

**-2 -18 0**

**-2 -26 0**

**-2 -34 0**

**-2 -42 0**

**-10 -18 0**

**-10 -26 0**

**-10 -34 0**

**-10 -42 0**

**-18 -26 0**

**-18 -34 0**

**-18 -42 0**

**-26 -34 0**

**-26 -42 0**

**-34 -42 0**

**-3 -11 0**

**-3 -19 0**

**-3 -27 0**

**-3 -35 0**

**-11 -19 0**

**-11 -27 0**

**-11 -35 0**

**-19 -27 0**

**-19 -35 0**

**-27 -35 0**

**-4 -12 0**

**-4 -20 0**

**-4 -28 0**

**-12 -20 0**

**-12 -28 0**

**-20 -28 0**

**-5 -13 0**

**-5 -21 0**

**-13 -21 0**

**-6 -14 0**

**-42 -48 0**

**-35 -41 0**

**-35 -47 0**

**-41 -47 0**

**-28 -34 0**

**-28 -40 0**

**-28 -46 0**

**-34 -40 0**

**-34 -46 0**

**-40 -46 0**

**-21 -27 0**

**-21 -33 0**

**-21 -39 0**

**-21 -45 0**

**-27 -33 0**

**-27 -39 0**

**-27 -45 0**

**-33 -39 0**

**-33 -45 0**

**-39 -45 0**

**-14 -20 0**

**-14 -26 0**

**-14 -32 0**

**-14 -38 0**

**-14 -44 0**

**-20 -26 0**

**-20 -32 0**

**-20 -38 0**

**-20 -44 0**

**-26 -32 0**

**-26 -38 0**

**-26 -44 0**

**-32 -38 0**

**-32 -44 0**

**-38 -44 0**

**-7 -13 0**

**-7 -19 0**

**-7 -25 0**

**-7 -31 0**

**-7 -37 0**

**-7 -43 0**

**-13 -19 0**

**-13 -25 0**

**-13 -31 0**

**-13 -37 0**

**-13 -43 0**

**-19 -25 0**

**-19 -31 0**

**-19 -37 0**

**-19 -43 0**

**-25 -31 0**

**-25 -37 0**

**-25 -43 0**

**-31 -37 0**

**-31 -43 0**

**-37 -43 0**

**-6 -12 0**

**-6 -18 0**

**-6 -24 0**

**-6 -30 0**

**-6 -36 0**

**-12 -18 0**

**-12 -24 0**

**-12 -30 0**

**-12 -36 0**

**-18 -24 0**

**-18 -30 0**

**-18 -36 0**

**-24 -30 0**

**-24 -36 0**

**-30 -36 0**

**-5 -11 0**

**-5 -17 0**

**-5 -23 0**

**-5 -29 0**

**-11 -17 0**

**-11 -23 0**

**-11 -29 0**

**-17 -23 0**

**-17 -29 0**

**-23 -29 0**

**-4 -10 0**

**-4 -16 0**

**-4 -22 0**

**-10 -16 0**

**-10 -22 0**

**-16 -22 0**

**-3 -9 0**

**-3 -15 0**

**-9 -15 0**

**-2 -8 0**

## MiniSAT output:

```
WARNING: for repeatability, setting FPU to use double precision
============================[ Problem Statistics ]=============================
|                                                                             |
|  Number of variables:            49                                         |
|  Number of clauses:             490                                         |
|  Parse time:                   0.00 s                                       |
|                                                                             |
============================[ Search Statistics ]==============================
| Conflicts |          ORIGINAL         |          LEARNT          | Progress |
|           |    Vars  Clauses Literals |    Limit  Clauses Lit/Cl |          |
===============================================================================
===============================================================================
restarts              : 1
conflicts             : 0              (-nan /sec)
decisions             : 14             (0.00 % random) (inf /sec)
propagations          : 49             (inf /sec)
conflict literals     : 0              (-nan % deleted)
Memory used           : 18.00 MB
CPU time              : 0 s

SATISFIABLE

-1 -2 -3 -4 -5 6 -7 -8 -9 -10 11 -12 -13 -14 -15 16 -17 -18 -19 -20 -21 -22 -23 -24 -25 -26 -27 28 -29 -30 -31 -32 33 -34 -35 -36 -37 38 -39 -40 -41 -42 43 -44 -45 -46 -47 -48 -49 0
```

**Verdict: *SATISFIABLE***