

# Homework 4

Zachary Moring

October 3, 2024

Code for this assignment is at: <https://github.com/zpm-bu/cs511-formal-methods/blob/assignments/lean/Homework/hw4.lean>

**Exercise 1.** Write the WFF  $\psi_n$  and justify how it accomplishes its task.

---

The WFF  $\psi_n$  is precisely

$$\psi_n = \psi_n^{\text{row}} \wedge \psi_n^{\text{col}} \wedge \psi_n^{\text{diag1}} \wedge \psi_n^{\text{diag2}}.$$

We define a queen at position  $(i_1, j_1)$  to capture another queen at position  $(i_2, j_2)$  if the two queens are in the same row, column, diagonal, or antidiagonal; that is, if  $i_1 = i_2$ , if  $j_1 = j_2$ , if  $i_1 - j_1 = i_2 - j_2$ , or if  $i_1 + j_1 = i_2 + j_2$ .

Consider an  $n \times n$  board with  $n$  queens on it. Assume that each queen is in one of  $n - k$  rows,  $1 \leq k \leq n$ . The positions of the queens are  $(i_1, j_1), \dots, (i_n, j_n)$ . By the pigeonhole principle, since we distribute  $n$  queens among  $n - k < n$  rows, at least one row must contain at least two queens. Thus  $i_a = i_b$  for some  $a, b \in \{1, \dots, n\}$  and thus these two queens capture each other. Thus, if you distribute  $n$  queens among  $n - k$ ,  $1 \leq k \leq n$  rows, the board is not a solution. By the contrapositive, a solution must distribute the queens among  $n$  rows. Therefore, each queen gets her own row and thus  $\psi_n^{\text{row}}$  is true.

Likewise, this same argument demonstrates that each queen must have her own column, diagonal, and antidiagonal in a solution. Therefore,

$$\psi_n = \psi_n^{\text{row}} \wedge \psi_n^{\text{col}} \wedge \psi_n^{\text{diag1}} \wedge \psi_n^{\text{diag2}}.$$

**Exercise 2.** The formal definition of substitution can be simplified if every WFF is such that:

1. There is at most one **binding** occurrence of the same variable, and
2. A variable cannot have both **free** and **bound** occurrences.

Formalize this idea.

---

Let  $\psi$  be a WFF in which (1) there is at most one binding occurrence for the same variable and (2) no variable has both free and bound occurrences.

Condition (1) removes nesting by preventing the same variable from being used multiple times. Thus, our substitution rules do not need to track which ‘version’ of the  $x$  we are substituting. Condition (2) ensures that we can substitute ‘with confidence,’ since it eliminates the possibility of accidentally capturing a free variable in a bound substitution.

Thus, substitution in this case can be done by directly rewriting  $x$  as  $u$ . Because there is no danger of ambiguity, there is no danger of accidentally making an ‘incorrect’ substitution.