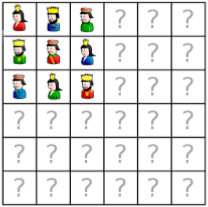
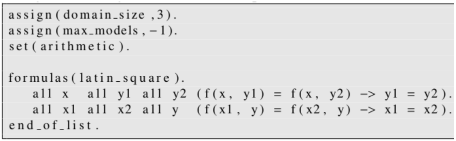
A short history of Sudoku

The game has European origins that can be traced back to 1779 to the Euler's *officer problem*. The problem asks you to arrange 36 officers in 6 × 6 grid, so that one officer from each of the six regiments appears in each row and column. This is a Latin square of order 6, whose search for solution led to developments in combinatorics In 1979, Howard Garns, a retired architect and puzzles fan, created the current version of the game under the name *Number Place*. The game was introduced in Japan in 1984 under the name Sudoku. In 1989, the first piece of software able to produce Sudoku grids appeared under the name DigiHunt. Sudoku has been the root of countless variants by adding new constraints, including: uniqueness value on diagonal, common cells between the regions, the digits within a region sum-up to a given value (i.e. Killer Sudoku), the sum of digits on columns and lines are restricted to some value (i.e. Kakuro).



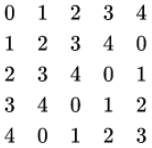
Example (Latin square)

An $n \times n$ array filled with n different values. Each value occurs exactly once in each row and exactly once in each column. How many Latin squares are there for $n = 3$?



```
assign(domain_size,3).
assign(max_models,-1).
set(arithmetic).

formulas(latin_square).
   all x   all y1  all y2  (f(x, y1) = f(x, y2) -> y1 = y2).
   all x1  all x2  all y   (f(x1, y) = f(x2, y) -> x1 = x2).
end_of_list.
```



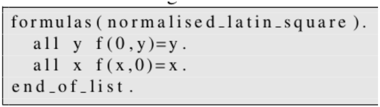
0	1	2	3	4
1	2	3	4	0
2	3	4	0	1
3	4	0	1	2
4	0	1	2	3

Example (Normalised Latin square)

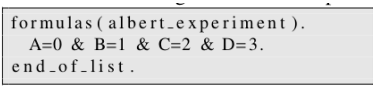
Both its first row and its first column are in their natural order (i.e. 0, 1, 2). How many normalised Latin squares are for $n = 3$? What about $n = 4$?

Example (A practical application of Latin squares)

Albert is a scientist that wants to test four different drugs (called A, B, C, and D) on four volunteers. He decides that every volunteer has to be tested with a different drug each week, but no two volunteers are allowed the same drug at the same time.



```
formulas(normalised_latin_square).
  all y  f(0,y)=y.
  all x  f(x,0)=x.
end_of_list.
```



```
formulas(albert_experiment).
  A=0 & B=1 & C=2 & D=3.
end_of_list.
```