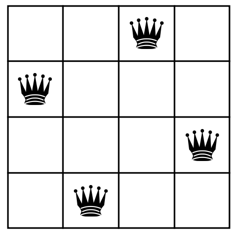
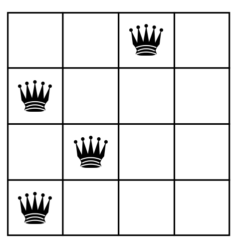
TD n°1

Exercise 1 (N-queens)

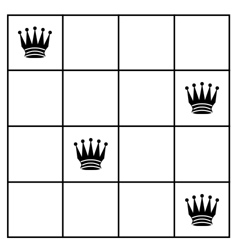
The N-queens problem consists in placing N queens on a chessboard $N \times N$ such that no two queens attack each other. A solution to the 4-queens problem is depicted below.



1. Define a model with binary variables.
2. Define a model with integer variables.
3. For both models, define the assignment corresponding to the solution in the given example and calculate the size of the search space.
4. Apply the backtracking algorithm to the 3-queens problem.
5. Apply the forward checking algorithm to the 3-queens problem. Compare with the result of the previous question.
6. Apply the forward checking algorithm to the 4-queens problem using different variable selection strategies.
7. Define and apply a local search algorithm to the 4-queens problem from this assignment.



8. Same question for this assignment.



Exercise 2 (Send more money)

The problem is to assign different digits to letters such that the following addition holds true:

$$\begin{array}{cccccc} & & S & E & N & D \\ + & & M & O & R & E \\ \hline = & M & O & N & E & Y \end{array}$$

Define a model of this problem.

Exercise 3 (Langford sequence)

A Langford sequence is a sequence of $2n$ integers $1, 1, 2, 2, \ldots, n, n$ such that for each number k in $1..n$, the two occurrences of k are separated by k numbers. For instance 4 1 3 1 2 4 3 2 is a Langford sequence with $n = 4$ (4 numbers 1 3 1 2 between the two 4, etc.).

Define a model of this problem.

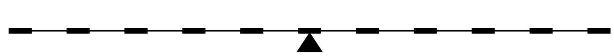
Exercise 4 (Zebra puzzle)

Five men with different nationalities and with different jobs live in consecutive houses on a street. The houses are painted different colors. The men have different pets and have different favorite drinks. Determine who owns a zebra and whose favorite drink is mineral water given the following clues.

1. The Englishman lives in the red house.
2. The Spaniard owns a dog.
3. The Japanese man is a painter.
4. The Italian drinks tea.
5. The Norwegian lives in the first house on the left.
6. The green house is on the right of the white one.
7. The photographer breeds snails.
8. The diplomat lives in the yellow house.
9. Milk is drunk in the middle house.
10. The owner of the green house drinks coffee.
11. The Norwegian's house is next to the blue one.
12. The violinist drinks orange juice.
13. The fox is in a house next to that of the physician.
14. The horse is in a house next to that of the diplomat.

Exercise 5 (balance)

Consider a foot seasaw with an odd number of seats. Anne, Lise and Max (weighing respectively w_a, w_l, w_m) want to sit such that they balance and p empty seats separate Lise and Anne.



Exercise 6 (Latin square)

A latin square is an array of size $n \times n$ filled with numbers $1..n$ such that each number occurs exactly once in each column and each row.

1

1	2
2	1

1	2	3
2	3	1
3	1	2

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

1	2	3	4	5
2	3	4	5	1
3	4	5	1	2
4	5	1	2	3
5	1	2	3	4

Define a model of the latin square.

Exercise 7 (Sudoku)

A sudoku puzzle is a latin square of size $n^2 \times n^2$ divided into regions of size $n \times n$ such that each region contains exactly one of each element of the set $1..n^2$. A problem and its solution are depicted below.

	6				1		2	
9	7		8	2		4		
	3	5			4			1
6		4					1	8
		7				2		
8	2					6		5
7			9			1	3	
		2		6	7		9	4
	4		5				8	

$\rightarrow$

4	6	8	3	5	1	9	2	7
9	7	1	8	2	6	4	5	3
2	3	5	7	9	4	8	6	1
6	9	4	2	7	5	3	1	8
5	1	7	6	8	3	2	4	9
8	2	3	4	1	9	6	7	5
7	5	6	9	4	8	1	3	2
3	8	2	1	6	7	5	9	4
1	4	9	5	3	2	7	8	6

1. Define a model of the sudoku problem.
2. Discuss propagation rules for contracting domains.
3. How to use CP in order to generate Sudoku problems?

Exercise 8 (slot assignment)

The organizers of a conference have to allocate N slots to N speakers such that each speaker is available only for a subset of slots (see table below, $N = 6$).

Speaker	Slots
1	$3, 4, 5, 6$
2	$3, 4$
3	$2, 3, 4, 5$

Speaker	Slots
4	$2, 3, 4$
5	$3, 4$
6	$1, 2, 3, 4, 5, 6$

Propose a model for this problem.

TP n°1

The goal is to implement a mini solver for finite domain CSPs. This work is scheduled for three laboratory sessions (plus personal work). Every team should be composed of four people. Any programming language can be used.

Complete search

The following (partial) specifications may guide the solver design, given an integer $n \geq 1$.

1. A *domain* is a subset of the set of integers $\{1, 2, \ldots, n\}$.

2. A *node* contains a sequence of n domains.

3. A *problem* defines a constraint satisfaction problem with n variables. A *problem* must implement the following operations.

Operation	Returns	Description
`initialNode()`	`Node`	Creates a node with the initial domains
`testSat(Node node)`	`Proof`	Tests the satisfaction of a node
`printSolution(Node node)`	`void`	Prints the solution in a node

   The `Proof` type may represent values for a failure (a domain is empty or a constraint is violated), a success (every domain is reduced to one value and all the constraints are satisfied) or a middle node (not yet a solution or a conflict).

4. A *branching strategy* selects a variable in a given node for a given problem and creates sub-nodes.

5. A *backtracking algorithm* is associated to a problem and a branching strategy. The following two operations are required.

Operation	Returns	Description
`solve()`	`int`	Solves the problem and returns the number of solutions
`branch(Node node)`	`void`	Creates sub-nodes of a node

Local search

The goal here is to implement an efficient local search algorithm for the N-queens problem and to compare it with a complete search algorithm. To this end, not all variable assignments have to be considered but only permutations in order to avoid that several queens appear in the same row or column. Moreover, generating a good initial assignment is very important.

Submission of documents

An archive containing the source code and a small report (maximum 10 pages) preferably written in LaTeX and compiled in PDF must be uploaded to **madoc** no later than **exactly one week after the third Lab session**. The report should present the solver design and report a result analysis (at least for the N-queens problem).