Foundations of Constraint Programming
Tutorial 1 (on October 16th)

Lukas Schweizer
WS 2017

Exercise 1.1:
Consider the task of assigning to each node of a finite graph a color in such a way that no two adjacent nodes have the same color. Such an assignment is called a coloring of the graph. Formulate the problem of given only 3-colors as constraint satisfaction problem.

Exercise 1.2:
Consider the depicted graph (to the right), consisting of 7 nodes to which the numbers 1, . . . , 7 must be assigned. Each of the nodes has at least one neighbor, whereas the sum of all neighbors of a node must be the sum given in the following table. E.g. if 3 is the assigned node value, then its neighbors must sum up to 14.

<table>
<thead>
<tr>
<th>node value</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>sum of its neighbors</td>
<td>20</td>
<td>18</td>
<td>14</td>
<td>19</td>
<td>2</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

Formulate the riddle as CSP.

Exercise 1.3:
A coloring of a graph involving the minimal number of colors is called the chromatic number of the graph. Formulate the problem of finding the chromatic number of a graph as a constrained optimization problem.

Exercise 1.4:
Formulate the following problem as a constrained optimization problem: Place a minimum number of queens on the chess board so that each unoccupied field comes under attack.