G53CLP
Constraint Logic Programming

Dr Rong Qu

Modeling CSPs – Case Study II
Solving Sudoku

- **Variables**
  - $\text{pos}_{i,j}$: number placed at position $(i,j)$

- **Domain**
  - $\text{pos}_{i,j} = \{1, \ldots, 9\}$
Solving Sudoku

- Constraints
  - \{1, \ldots, 9\} on each row
  - \{1, \ldots, 9\} on each column
  - \{1, \ldots, 9\} in each 3X3 square
Solving Sudoku

- **Constraints**
  1. All different each row
  2. All different each column
  3. All different each 3X3 square

- How to model constraint 3?
Solving Sudoku

- **Constraints**
  - All different pos[i,j] = \{1, ..., 9\}
  - ?
Solving Sudoku – Demo
Data File & Project in OPL

- Input data in OPL

- Project in OPL
  - Add/insert model file .mod
  - Add/insert data file .dat
  - Data type declared in model file .mod will look up the data file .dat
Solving Sudoku in OPL

//.dat file
input=
[[0 6 0 0 5 0 0 2 0]
 [0 0 0 3 0 0 0 9 0]
 [7 0 0 6 0 0 0 1 0]
 [0 0 6 0 3 0 4 0 0]
 [0 0 4 0 7 0 1 0 0]
 [0 0 5 0 9 0 8 0 0]
 [0 4 0 0 0 1 0 0 6]
 [0 3 0 0 0 8 0 0 0]
 [0 2 0 0 4 0 0 5 0]];}
Solving Sudoku in OPL

//.mod file
range dim1 1..9;
range dim2 1..9;
range bindex 0..2;
int input[dim1,dim2] = ...; //read in the .dat

?
Solving Sudoku in OPL

solve{
    forall(i in 1..9)
        forall(j in dim2 : input[i,j] <> 0)
            pos[i,j] = input[i,j] ;

    ...

};
Solving Sudoku in OPL

solve{

... 
forall(j in dim2)
    alldifferent(all(i in dim1) pos[i,j]);

forall(i in dim1)
    alldifferent(all(j in dim2) pos[i,j]);

...

};
Solving Sudoku in OPL

solve{
  ...
  ...
  forall(b1 in bindex){
    forall(b2 in bindex){
      ?
    }
  }
};
Exercise – Coins in Bank

- You have a bag of coins to save in your bank
- Weights of coins

<table>
<thead>
<tr>
<th>Weight (g)</th>
<th>£1</th>
<th>50p</th>
<th>20p</th>
<th>10p</th>
<th>5p</th>
<th>2p</th>
<th>1p</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

- If the bag of coins weighs 131g, how much in total?
- How about 4g? How many possible solutions?
- How can you solve this problem in CP?
Exercise – Magic Sequence

- A magic sequence of length $n$ is a sequence of integers $\{X_0, \ldots, X_{n-1}\}$, such that for all $i = 0, \ldots, n-1$, the number $i$ occurs exactly $X_i$ times in the sequence.

- For example, the sequence $[6,2,1,0,0,0,1,0,0,0]$ is a magic sequence of length $n=10$, since 0 occurs 6 time in it, 1 occurs twice, 2 occurs once, and 3 doesn't occurs (hence it's zero), ....
Lab Session next Week ...

In IBM ILOG OPL IDE

- I: build project with data input
- II: solve the Sudoku instances

- Supplement exercises for those who are interested to learn more