

## Aufgabenblatt 3

### Aufgabe 1 (10 %)

python\_constraint ist sehr kleines Python-Modul, um CSPs (constraint satisfaction problems) zu lösen:

<https://github.com/python-constraint/python-constraint>

Machen Sie sich mit dem Modul vertraut, indem Sie folgende in der github vorhandenen Programm-Beispiele durchgehen und ausführen:

- Sodoku
- Kryptogramm SEND + MORE = MONEY
- Zebrarätsel (im github *Einstein* genannt);  
siehe auch [https://de.wikipedia.org/wiki/Zebra\\_rätsel](https://de.wikipedia.org/wiki/Zebra_rätsel)

### Aufgabe 2 (20 %)

Lösen Sie das in der Vorlesung besprochene Beispiel *Timetabling* mit python\_constraint.

### Aufgabe 3 (30 %)

Formulieren Sie die Kartenfärbung der 16 deutschen Bundesländer als CSP und suchen Sie eine Färbung mit python\_constraint. Genügen 3 Farben oder werden 4 Farben benötigt?



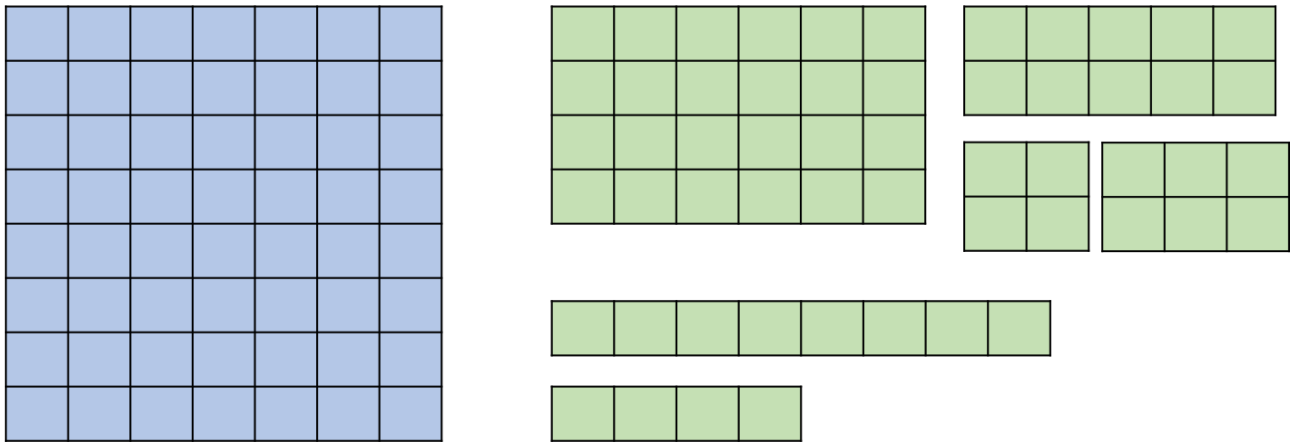
## Aufgabe 4 (Packungsproblem)

(40 %)

Ein Packungsproblem (hier 2-dimensional) besteht in der Aufgabe N kleinere Rechtecke in ein großes Rechteck überschneidungsfrei zu platzieren. Ein Rechteck ist dabei ein  $n \times m$  großes Quadratgitter. Die Rechtecke werden entweder horizontal oder vertikal am Gitter ausgerichtet platziert.

Wie lässt sich das Packungsproblem ganz allgemein als CSP formulieren?

Lösen Sie das hier abgebildete Packungsproblem mit `python_constraint`.



Die  $N = 6$  grünen Rechtecke ( $6 \times 4$ ,  $8 \times 1$ ,  $4 \times 1$ ,  $5 \times 2$ ,  $2 \times 2$  und  $3 \times 2$ ) sollen in das blaue Rechteck ( $7 \times 8$ ) überschneidungsfrei platziert werden.