

Aufgabe 1

- a) Per Hand können wir die dritte Gleichung in die ersten beiden einsetzen, wodurch sich $6 \cdot \text{Männer} + \text{Kinder} = 100$ und $13 \cdot \text{Männer} + 0,5 \cdot \text{Kinder} = 100$ gibt. Ziehen wir letztere doppelt von der ersteren ab, ergibt sich $-20 \cdot \text{Männer} = -100$, also erhalten wir **fünf Männer, 25 Frauen und 70 Kinder**. Dasselbe spuckt folgender Code aus:

```
var 1..100: maenner;
var 1..100: frauen;
var 1..100: kinder;

constraint maenner + frauen + kinder = 100;
constraint 3*maenner + 2*frauen + 0.5*kinder = 100;
constraint frauen = 5*maenner;

solve satisfy;

output [show(maenner), " Männer, ", show(frauen), " Frauen und ", show(kinder),
" Kinder"]
```

- b) Wir setzen eine Variable gleich 1, falls die betreffende Person schuldig ist, und sonst 0. Aus den acht möglichen Kombinationen (Tick, Trick, Track) können wir durch Ticks Aussage (0, 0, 1), (1, 0, 1), (0, 1, 0) und (1, 1, 0) streichen. Durch Tricks Aussage können wir nur (1, 0, 0) streichen, da (1, 0, 1) bereits weg ist. Es bleiben (0, 0, 0), (0, 1, 1) und (1, 1, 1). Da laut Track Trick und Track nicht beide schuldig sind, fallen die letzten beiden weg, so dass nur noch (0, 0, 0) übrig bleibt. Demensprechend kann Donald **alle drei freisprechen**. Dasselbe spuckt folgender Code aus:

```
var 0..1: tick;
var 0..1: trick;
var 0..1: track;

constraint trick = track;
constraint tick <= track;
constraint trick + track < 2;

solve satisfy;

output ["1 = schuldig, 0 = unschuldig: Tick ", show(tick), ", Trick ",
show(trick), ", Track ", show(track)]
```

- c) Wir setzen eine Variable gleich 1, falls die betreffende Person ein Ritter ist, und sonst 0. Da in der ersten Teilaufgabe alle drei Aussagen disjunkt sind, kann maximal eine der drei Aussagen korrekt sein. Sind alle Aussagen falsch, wären alle drei Schurken. Dann hätte C aber die Wahrheit gesagt. Demensprechend ist genau eine Aussage korrekt. Dies ist jedoch gleichbedeutend, dass genau ein Ritter auf der Insel lebt. Da

dies der Aussage von B entspricht, ist **B Ritter und die anderen beiden Schurken.**

Dasselbe spuckt folgender Code aus:

```
var 0..1: A;
var 0..1: B;
var 0..1: C;

constraint if A==1 then A+B+C = 2 else A+B+C != 2 endif;
constraint if B==1 then A+B+C = 1 else A+B+C != 1 endif;
constraint if C==1 then A+B+C = 0 else A+B+C != 0 endif;

solve satisfy;

output ["1 = Ritter, 0 = Schurke: A ", show(A), ", B ", show(B), ", C ",
show(C)]
```

Die zweite Teilaufgabe läuft ähnlich. Wiederum sind alle Aussagen disjunkt, so dass maximal eine Aussage richtig sein kann. Sind alle drei Aussagen falsch, wären alle drei Schurken. Dann hätte auch jeder gelogen, so dass **„alle drei Schurken“** eine mögliche Lösung ist. Ist genau eine Aussage richtig, gibt es genau einen Ritter. Dies entspricht der Aussage von A, so dass dann **A Ritter und die anderen beiden Schurken** sind. Folgender Code findet zunächst die erstgenannte Lösung:

```
var 0..1: A;
var 0..1: B;
var 0..1: C;

constraint if A==1 then A+B+C = 1 else A+B+C != 1 endif;
constraint if B==1 then A+B+C = 2 else A+B+C != 2 endif;
constraint if C==1 then A+B+C = 3 else A+B+C != 3 endif;

solve satisfy;

output ["1 = Ritter, 0 = Schurke: A ", show(A), ", B ", show(B), ", C ",
show(C)]
```

In diesem Fall bleibt die Identität von A also ungeklärt.