

Arbeitsblatt 3

Das Überschreiben von Methoden

Die neue Aufgabe ist identisch mit der vom 2. Arbeitsblatt, aber: Es ist wieder Feierabend und es muss aufgeräumt werden. Dabei können mehrere Gegenstände auf einem Feld liegen.

Bei der klassischen Programmierung wird nun der **Algorithmus abgeändert** und durch mindestens einen Unteralgorithmus ergänzt.

Nun die Eleganz der OOP:

Wir verändern den Algorithmus nicht, sondern benutzen nur eine **andere Roboterklasse (RaeumRoboter)**, bei der „vor“ eine andere Bedeutung hat. Das geht, obwohl wir die Programmierung von „vor“ nicht kennen, sie ist geheim und liegt nicht im Quelltext vor (Bestandteil vom Niki-System)!

aus „vor“ wird (umgangssprachlicher Algorithmus):

die neue Methode „vor“:

vor (des Beerbtens)
solange auf dem Feld etwas liegt, nimm*

Syntax für Aufruf der ursprünglichen Methode „vor“:

| Python (Roboter wird beerbt) | Java (irgendwer wird beerbt) |
|-------------------------------------|-------------------------------------|
| Roboter.vor(self) | super.vor() |

Programmieren Sie die Klasse RaeumRoboter und lösen Sie die Aufgabe!

- * kann als zusätzliche (private) Methode programmiert werden.
- Testen Sie das Programm u.a. in einer Welt, in der alle Felder (bis auf das Startfeld) mit Werkzeugen belegt ist.