

```
# gem install gosu
```

Falls gefragt wird, welches gosu-Paket installiert werden soll, wählt man am besten **gosu 0.7.14 (ruby)** aus.

Falls eine Fehlermeldung der Art

```
ERROR: While executing gem ... (Gem::\nGemNotFoundException)\n    Could not find gosu (> 0) in any repository
```

kommt, sollte man die Installation einfach wiederholen, bis das Paket gosu gefunden wird.

Nach der Kompilierung von Gosu kann man die Entwicklerpakete wieder deinstallieren. Ruby & Co. müssen natürlich installiert bleiben.

GUI-START

Die grafische Oberfläche besteht aus zwei Teilen. Zum einen aus der Oberfläche für alle Einstellungen des Spiels **robots-gui-helper** und zum anderen aus der Oberfläche für das Spielbrett **robots-gui**. Da die **robots-gui** nie direkt gestartet werden sollte, werden dessen Optionen hier nicht näher erläutert.

Den GUI-Starter ruft man per

```
$ ./robots-gui-helper
```

auf. Im Standardfall kann man nun bereits auf „*Spiel starten*“ klicken und die Referenz-KI sollte den Roboter bewegen.

Die Einstellungen im Start sind alle selbsterklärend. Es soll hier nur auf die Optionen eingegangen werden, die nicht offensichtlich sind.

Über „*Neues Spiel starten*“ kann man einstellen, ob der Roboter von seiner letzten Position aus der Datei **bot.txt** oder von der Startposition auf dem Spielbrett starten soll. Zusätzlich wird das Spielfeld und der Kartenstapel nicht neu eingelesen, wenn der Haken nicht gesetzt ist. So kann man ein unterbrochenes Spiel fortsetzen.

Entfernt man den Haken bei „*Spiel animieren*“, läuft der Roboter nicht über das Spielfeld, sondern das Spiel wird bis zum Ende berechnet, der Roboter also das Ziel gefunden hat oder zerstört wurde. Erst dann wird die Anzeige freigeschaltet und man sieht den Roboter an der Endposition.

Mit „*Spiel starten*“ startet man das Spiel und kann dem Roboter bei der Bewegung zuschauen. Der Roboter hinterlässt eine transparente grüne Spur (Nein, er verliert kein Öl!), mit der man nachvollziehen kann, welche Felder er wie häufig