Lektion 4: Der Rover soll fahren! — Übung 2: Erweiterung der Fahrbefehle

Den vier Fahrbefehlen aus Übung 1 fehlten die abschließenden Anführungszeichen und die schließende Klammer, weil es weitere Möglichkeiten gibt, um die Befehle zu vervollständigen. In dieser Lektion sollen diese Möglichkeiten untersucht werden.

Lernziele:

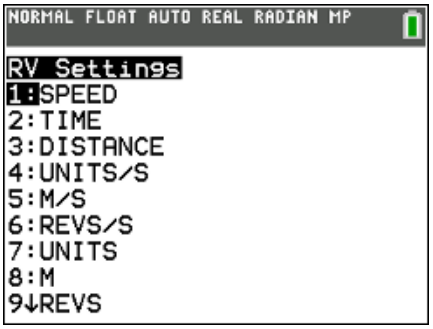
- Erweiterung der Fahrbefehle des Rovers
- Optionen für **RIGHT** und **LEFT**
- Der Befehl **Wait**

Parameter für FORWARD und BACKWARD

Diese Fahrbefehle haben drei optionale Parameter:

- **SPEED** (Geschwindigkeit)
- **TIME** (Zeit)
- **DISTANCE** (Entfernung)

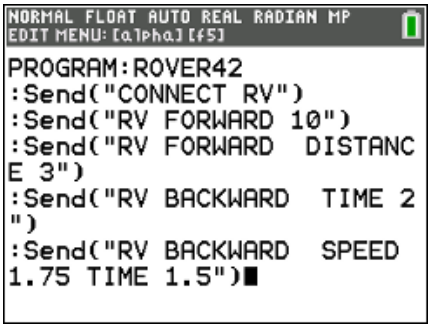
Man findet sie im Menü **prgm > Hub > Rover (RV)… > RV Settings…** ebenso wie die Einheiten für die Geschwindigkeit **UNITS/S** and **M/S.**



Verwendung von **DISTANCE-T-T**, **SPEED** und **TIME**

Beispiele für unterschiedlichen Anwendungen im Befehl **FORWARD:**

- **FORWARD DISTANCE** # ist gleichbedeutend mit **FORWARD** #.
  - Die Voreinstellung für die Entfernungseinheit # ist 10 cm.
- **FORWARD DISTANCE # M** bewegt den Rover um # Meter.
- **FORWARD # SPEED #** stellt zusätzlich die Geschwindigkeit ein.
  - Speed darf einen Wert zwischen 1.4 und 2.3 haben. Werte außerhalb dieses Bereichs erzeugen eine Fehlermeldung des Hubs.
- **FORWARD TIME** #.

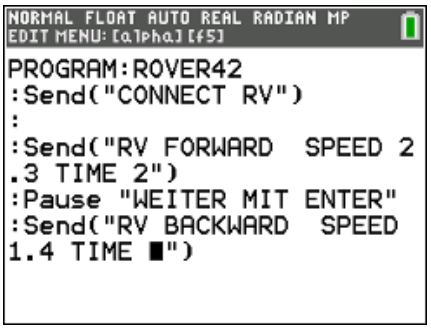


Man kann höchstens 2 der drei Optionen in den Befehlen **FORWARD** und **BACKWARD** verwenden, da die dritte von den beiden anderen abhängig ist. Sollte der anzuwendende Wert in einer Variablen gespeichert oder aus einem Term berechnet werden, so muss **eval( )** verwendet werden.

SPEED und TIME Programm

1. Das rechts dargestellte Programm enthält den Befehl:
   Send("RV FORWARD SPEED 2.3 TIME 2")

2. Nun soll der richtige Wert für **TIME** eingefügt werden, so dass der Rover wieder in seine Ausgangsposition kommt:
   Send("RV BACKWARD SPEED 1.4 TIME *?* **")**



Hinweist: **DISTANCE = SPEED * TIME**

Optionen für RIGHT und LEFT

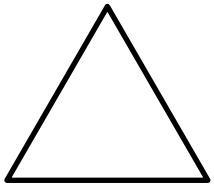
Die Voreinstellung für diese Befehle ist eine Drehung um 90° rechts- (**RIGHT**) oder linksherum (**LEFT**), aber man kann auch einen Winkel einfügen, sodass der Rover sich um jeden Winkel zwischen -360° … 360° dreht. Negative Werte sind erlaubt; d.h., **LEFT -90** ist dasselbe wie **RIGHT 90**.

3. Im Beispiel wurde ein Befehl eingefügt, durch den der Rover rechtsherum um 135° dreht: **RV RIGHT 135.** Man muss die Zahl **135** eingeben und auch die abschließenden Anführungszeichen und die Klammer. Die Maßeinheit **DEGREES** ist nicht notwendig, aber sie ist verfügbar im Menü **RV Settings…** , wenn man die Drehung verdeutlichen will.

```
NORMAL FLOAT AUTO REAL RADIAN MP
EDIT MENU: [alpha] [f5]
PROGRAM:ROVER42
:Send("CONNECT RV")
:
:Send("RV RIGHT 135")
:
:Send("RV RIGHT 45 DEGREES
")
:
:Send("RV LEFT eval(π) RAD
IANS")
```

Man kann die Drehung auch in **RADIANS** oder **GRADS** angeben (ebenfalls im Menü **RV Settings…**), aber dann müssen diese Maßeinheiten auch zwingend dazu geschrieben werden.

Im Beispielprogramm rechts sind ein paar Befehle aufgelistet. Wie genau steht der Rover, wenn diese drei Befehle ausgeführt sind?

Es soll ein Programm geschrieben werden, so dass der Rover ein gleichseitiges Dreieck fährt. Man kann mit den folgenden Befehlen anfangen und vielleicht auch eine Schleife verwenden:

Send(“CONNECT RV”)
Send(“RV FORWARD ?”)
Send(“RV LEFT ?”