

# Der Edison-Roboter

Informationen zur Inbetriebnahme und Programmierung des Edison-Roboters

- [Den Edison in Betrieb nehmen](#)
- [Erste Programme mit dem Edison](#)
- [Programmierung mit Ed-Python](#)
- [Projekte mit Edison](#)
- [Fortgeschrittene Programmierung](#)

# Den Edison in Betrieb nehmen

## Batterien einlegen und testen

[Get-started-with-Edison.pdf](#)

[Edscratch](#)

## Kalibrieren der Edisons nach Anleitung

Der Edison funktioniert besser, wenn er vor Gebrauch kalibriert wird.

## Testen und Kalibrieren von Sensoren

## Hinderniserkennung

Teste die Sensoren im Edison. Schreibe ein Programm, das eine Nachricht gibt, wenn der Edison ein Hindernis bemerkt.

## Algorithmus

1. Edison ist stationär
2. Die Hinderniserkennung wird eingeschaltet.

3. Wenn ein Hindernis (Hand, Mäppchen vor den Edison bewegt wird, dann sollen die LEDs aufleuchten.
4. Ist das Hindernis weg, dann gehen die LEDs aus.

## erweiterter Algorithmus

Je nachdem ob das Hindernis rechts oder links ist, gehen die entsprechenden LEDs an und aus.

# Linienverfolger

Hierfür wird ein Testpapier benötigt.

1. Der Edison ist stationär
2. Die Linienverfolgung wird eingeschaltet
3. Wenn ein Blatt Papier mit weißen und schwarzen Stellen unter dem Edison bewegt wird, geben die LEDs Meldung über den Zustand der Linienerkennung

# Erste Programme mit dem Edison

## Hindernis umfahren

Basierend auf dem Beispielprogramm "avoid\_obstacles" verändere die Reaktion des Edison auf eine Hinderniserkennung.

Wer das Arbeiten mit Edscratch zu langweilig findet, kann übergehen zu [Edpy](#).

# Programmierung mit Ed-Python

Die Programmierung des Edison erfolgt auf der Seite [Edpy](#). Dort gibt es Beispielprogramme und eine Anleitung für jede einzelne Funktion in Edpy.

## Analyse von Beispielcode

Schauen wir uns für den Anfang einmal den Beispielcode für die Hinderniserkennung und -vermeidung an. Die Erklärung die einzelnen Schritte sind in den Kommentaren im Code.

```
#-----Setup-----

import Ed

Ed.EdisonVersion = Ed.V2

Ed.DistanceUnits = Ed.CM
Ed.Tempo = Ed.TEMPO_MEDIUM

#-----Your code below-----
# Bis hierhin sollte nichts an dem Code verändert werden.

# Der Edison erkennt Hindernisse mithilfe von Infrarotlicht. Dazu hat der Edison eine Infrarot-LED vorne, die mit
diesem Befehl eingeschaltet wird.
Ed.ObstacleDetectionBeam(Ed.ON)
Diese beiden Variablen werden später verwendet, um zu bestimmen, ob sich der Edison rechts oder links herum
drehen soll und ob ein Hindernis erkannt wurde.
#set up variables
SideTurn = 0
obstacle = 0

#Diese Schleife läuft unendlich lange. Bis das Programm mit der Quadrattaste beendet wird.
while True:
```

```

[]# Der Edison fährt mit halber Geschwindigkeit. Die drei Parameter bedeuten:
    # Ed.FORWARD: eigentlich selbsterklärend.
    # Ed.Speed_5: Die Geschwindigkeit, mit dem der Edison fahren soll. Es gibt die Geschwindigkeitsstufen 1-10.
    # Ed.DISTANCE_UNLIMITED: Der Edison fährt solange, bis das Programm etwas anderes sagt.
[]Ed.Drive(Ed.FORWARD, Ed.SPEED_5, Ed.DISTANCE_UNLIMITED)
[]# Nachdem der Edison anfang zu fahren, wird jetzt gemessen, ob ein Hindernis im Weg ist. Das Ergebnis wird in
einer der folgenden Variablen gespeichert:
    # Ed.OBSTACLE_NONE: Kein Hindernis erkannt
    # Ed.OBSTACLE_RIGHT: Hindernis an der rechten Seite
    # Ed.OBSTACLE_LEFT: Hindernis an der linken Seite
    # Ed.OBSTACLE_AHEAD: Hindernis voraus
    # Diese Variablen einhalten Zahlen Ed.OBSTACLE_NONE ist die Nummer 0. Daher funktioniert folgende
Abfrage:
[]obstacle = Ed.ReadObstacleDetection()
[]# Wenn eine Zahl > 0 zurückgegeben wird, ist ein Hindernis erkannt worden. Egal wo.
[]if obstacle>Ed.OBSTACLE_NONE:
[]# Beide LED gehen an und der Edison fährt mit halber Geschwindigkeit 3cm zurück.
[]Ed.LeftLed(Ed.ON)
[]Ed.RightLed(Ed.ON)
[]Ed.PlayBeep()
[]Ed.Drive(Ed.BACKWARD, Ed.SPEED_5, 3)
[]# Hier wird überprüft, in welcher Richtung das Hindernis liegt. Der Edison dreht sich in die andere Richtung,
oder, wenn das Hindernis voraus ist, zufällig rechts oder links.
[]if obstacle==Ed.OBSTACLE_LEFT:
[]Ed.Drive(Ed.SPIN_RIGHT, Ed.SPEED_5, 90)
[]elif obstacle==Ed.OBSTACLE_RIGHT:
[]Ed.Drive(Ed.SPIN_LEFT, Ed.SPEED_5, 90)
    # Hindernis voraus.
[]elif obstacle==Ed.OBSTACLE_AHEAD:
    # Die Variable SideTurn ist beim ersten Durchlauf 0. Daher tritt der zweite Fall ein.
[]if SideTurn ==1:
[]Ed.Drive(Ed.SPIN_RIGHT, Ed.SPEED_5, 90)
[]else:
[]Ed.Drive(Ed.SPIN_LEFT, Ed.SPEED_5, 90)
[]#after the obstacle has been avoided turn the LEDs off
[]Ed.LeftLed(Ed.OFF)
[]Ed.RightLed(Ed.OFF)
[]# Bei jedem Durchlauf der Schleife wird der Wert von SideTurn geändert, so dass sich eine zufällige
Drehrichtung ergibt.
[]if SideTurn==1:

```

```
    SideTurn=0
```

```
  else:
```

```
    SideTurn=1
```

Darüber hinaus gibt es noch andere Funktionen:

- Linienverfolger,
- Lichtsensor

# Projekte mit Edison

Folgende Vorschläge können als Anregung oder Anleitung für eigene Projekte dienen:

- [Linienverfolgertest](#)
- [Candy Dispenser](#)
- [EdPrinter](#)
- [EdDigger](#)
- [EdTank](#)
- [Einführung in Edison](#)
- [EdPrinter](#)
-



# Fortgeschrittene Programmierung

## Programmieren mit EventHandlern

EventHandler geben uns die Möglichkeit, eine Reaktion auf bestimmte Ereignisse im Edison bei Programmbeginn zu registrieren, die dann das Programm unterbrechen um den vorher bestimmten Code auszuführen.