

## Lektion 4: Arbeitsblatt 4.1 - Fahren im Quadrat

In dieser Aufgabe musst du ein Programm schreiben, mit dem dein Edison-Roboter in Form eines Quadrats fahren kann.

### Du bist dran:

Schreibe ein Programm, damit dein Edison-Roboter beim Fahren ein Quadrat bildet. Verwende die Befehle, die du bereits gelernt hast, einschließlich `Ed.Drive()` und eine Variable. Achte darauf, dass dein Programm damit endet, dass dein Edison-Roboter an der gleichen Stelle steht, an der er gestartet ist.

Lade dein Programm herunter und teste es mit Hilfe des Aktivitätsblatt 4.1, indem du deinen Edison am Startpunkt platzierst und den Linien folgst. du kannst auch ein größeres Quadrat mit farbigem Klebeband erstellen, um die Linien und den Startpunkt auf einem Schreibtisch oder dem Boden zu markieren.

1. Wie sieht dein Programm aus? Schreibe deinen Code unten auf.

---

---

---

---

---

---

---

---

2. Wie viele Funktionsaufrufe hast du in diesem Programm verwendet?

---

3. Hast du irgendwelche doppelten Codezeilen in deinem Programm? Wenn ja, wie lauten sie und wie oft hast du sie verwendet?

---

---

---

---

## Lektion 4: Arbeitsblatt 4.2 - Eine Schleife verwenden, um im Quadrat zu fahren

In dieser Aufgabe musst du ein Programm schreiben, das es deinem Edison-Roboter ermöglicht, in Form eines Quadrats zu fahren.

Mit Hilfe von Arbeitsblatt 4.1 hast du ein Programm geschrieben, das dieselben Befehle mehrfach verwendet. Du musstest `Ed.Drive()` mit einem Richtungsparameter von `Ed.FORWARD` viermal verwenden, einmal für jede Seite des Quadrats. Außerdem musstest du `Ed.Drive()` mit einem Richtungsparameter von `Ed.SPIN_LEFT` viermal verwenden, um jede Ecke zu drehen.

Fandest du es langweilig, immer wieder die gleichen Befehle zu schreiben?

Dieselben Befehle immer wieder zu wiederholen, ist für einen Computer kein Problem, aber ein Programm auf diese Weise zu schreiben, ist nicht sehr effizient. Stattdessen ist es besser, eine Schleifenstruktur zu verwenden.

Wir können ein Programm schreiben, das Edison mit weniger Code in einem Quadrat fahren lässt, indem wir eine "for"-Schleife verwenden. Dadurch wird das Schreiben des Programms effizienter. Da wir weniger Codezeilen benötigen, wird durch die Verwendung der "for"-Schleife auch die Wahrscheinlichkeit von Tippfehlern und Syntaxfehlern im Programm verringert.

### Die 'for'-Schleife und die 'range()'-Funktion in Python

In Python ist eine "for"-Schleife eine Kontrollstruktur, die dazu verwendet werden kann, eine Reihe von Befehlen oder Anweisungen beliebig oft zu wiederholen.

Mit einer "for"-Schleife kannst du einen Anweisungsblock beliebig oft wiederholen (auch 'iterieren' genannt).

Die "for"-Schleife wird in Python oft mit der Funktion `range()` kombiniert. Die Funktion `range()` gibt eine Menge von Werten innerhalb eines bestimmten Bereichs zurück.

In EdPy hat `range()` nur einen Eingabeparameter. Dieser Eingabeparameter bestimmt die Obergrenze der Menge, und die Untergrenze ist immer 0.

Die Funktion `range()` liefert Werte von 0 bis (Eingabeparameter - 1).

*Beispiel:*

`range(4)` gibt 4 Werte in der Menge: 0, 1, 2, 3 zurück.

Schauen wir uns ein Beispiel an:

```
15 ▾ for i in range(4):  
16     Ed.PlayBeep()  
17     Ed.TimeWait(1,Ed.TIME_SECONDS)
```

In diesem Beispiel wird die 'for'-Schleife viermal durchlaufen, wodurch die Variable 'i' die Werte 0, 1, 2 und 3 annimmt. Bei jedem Durchlauf führt die Schleife den Anweisungsblock bestehend aus `Ed.PlayBeep()` und `Ed.TimeWait()` aus.

Das Ergebnis? Der Piepton wird viermal mit einer Verzögerung von einer Sekunde zwischen den einzelnen Pieptönen abgespielt.

### Sie sind dran:

Schreibe ein Programm mit der 'for'-Schleife und der 'range()'-Funktion, so dass dein Edison-Roboter beim Fahren ein Quadrat bildet. Du solltest in der Lage sein, das Programm mit nur zwei `Ed.Drive()`-Funktionen zu erstellen, eine für die Vorwärtsfahrt und eine für die Drehung.

Vergiss nicht, innerhalb deiner Schleife einen Doppelpunkt und eine korrekte Einrückung einzufügen.

Lade dein Programm herunter und teste es mit Hilfe des Aktivitätsblatt 4.1. Setze deinen Edison an den Startpunkt und folge den Linien. Vergewissere dich, dass dein Programm mit dem Edison an derselben Stelle endet, an der es begonnen hat. Du kannst auch ein größeres Quadrat mit farbigem Klebeband erstellen, um die Linien und den Startpunkt auf einem Tisch oder dem Boden zu markieren.

1. Wie sieht dein Programm aus? Schreibe deinen Code unten auf.

---

---

---

---

---

---

---

## Lektion 4: Arbeitsblatt 4.3 - Einfahren eines Dreiecks und eines Sechsecks

Bei dieser Aufgabe musst du zwei verschiedene Programme schreiben, um deinen Edison-Roboter dazu zu bringen, in Form eines Dreiecks und dann in Form eines Sechsecks zu fahren.

**Sie sind dran:**

### Aufgabe 1: Fahre im Dreieck

Schreibe ein Programm, damit dein Edison-Roboter beim Fahren ein Dreieck bildet.

Lade dein Programm herunter und teste es mit Hilfe von Aktivitätsblatt 4.2, indem du deinen Edison am Startpunkt platzierst und den Linien folgst. Du kannst auch ein größeres Dreieck mit farbigem Klebeband erstellen, um die Linien und den Startpunkt auf einem Schreibtisch oder dem Boden zu markieren.

1. Wie oft wurde die "for"-Schleife für die Dreiecksform ausgeführt?

\_\_\_\_\_

### Aufgabe 2: Fahren eines Sechsecks

Schreibe ein Programm, damit dein Edison-Roboter beim Fahren ein Sechseck bildet.

Lade dein Programm herunter und teste es mit Hilfe von Aktivitätsblatt 4.3, indem du deinen Edison am Startpunkt platzierst und den Linien folgst. Du kannst auch ein größeres Sechseck mit farbigem Klebeband anfertigen, um die Linien und den Startpunkt auf einem Tisch oder dem Boden zu markieren.

2. Wie oft wurde die "for"-Schleife für die Sechseckform ausgeführt?

\_\_\_\_\_

3. Du solltest ein Muster zwischen der Anzahl der Seiten der Form und der Anzahl der Durchläufe der "for"-Schleife erkennen. Beschreibe dieses Muster.

\_\_\_\_\_  
\_\_\_\_\_

4. Wie oft müsste die "for"-Schleife ausgeführt werden, um eine regelmäßige (d. h. alle Seiten sind gleich) 12-seitige Form zu zeichnen?

\_\_\_\_\_

## Lektion 4: Arbeitsblatt 4.4 - Herausforderung! Im Kreis fahren

Bei dieser Aufgabe musst du ein Programm schreiben, das deinen Edison-Roboter dazu bringt, im Kreis zu fahren.

### Sie sind dran:

Schreibe ein Programm, in dem dein Edison-Roboter im Kreis fährt. Dein Edison muss in der Form eines Kreises fahren und sich nicht nur an einer Stelle drehen.

Lade dein Programm herunter und teste es mit Hilfe des Aktivitätsblatt 4.4, indem du deinen Edison am "Startpunkt" platzierst und der Linie folgst. du kannst deinen Roboter auch um ein beliebiges rundes Objekt fahren lassen, z. B. einen runden Mülleimer oder einen runden Tisch.

*Hinweis:* Eine Form mit vielen Hunderten von sehr kleinen Seiten kann einem Kreis sehr nahe kommen.

1. Wie oft muss die Schleife ausgeführt werden, um die Form zu einem Kreis zu machen?

---

---

---

2. Wie weit fährt dein Roboter jedes Mal, wenn du deine Schleife ausführst?

---

---

3. Fährt dein Roboter in einem perfekten Kreis? Wenn nicht, fällt dir ein Grund ein, warum nicht?

---

---

---

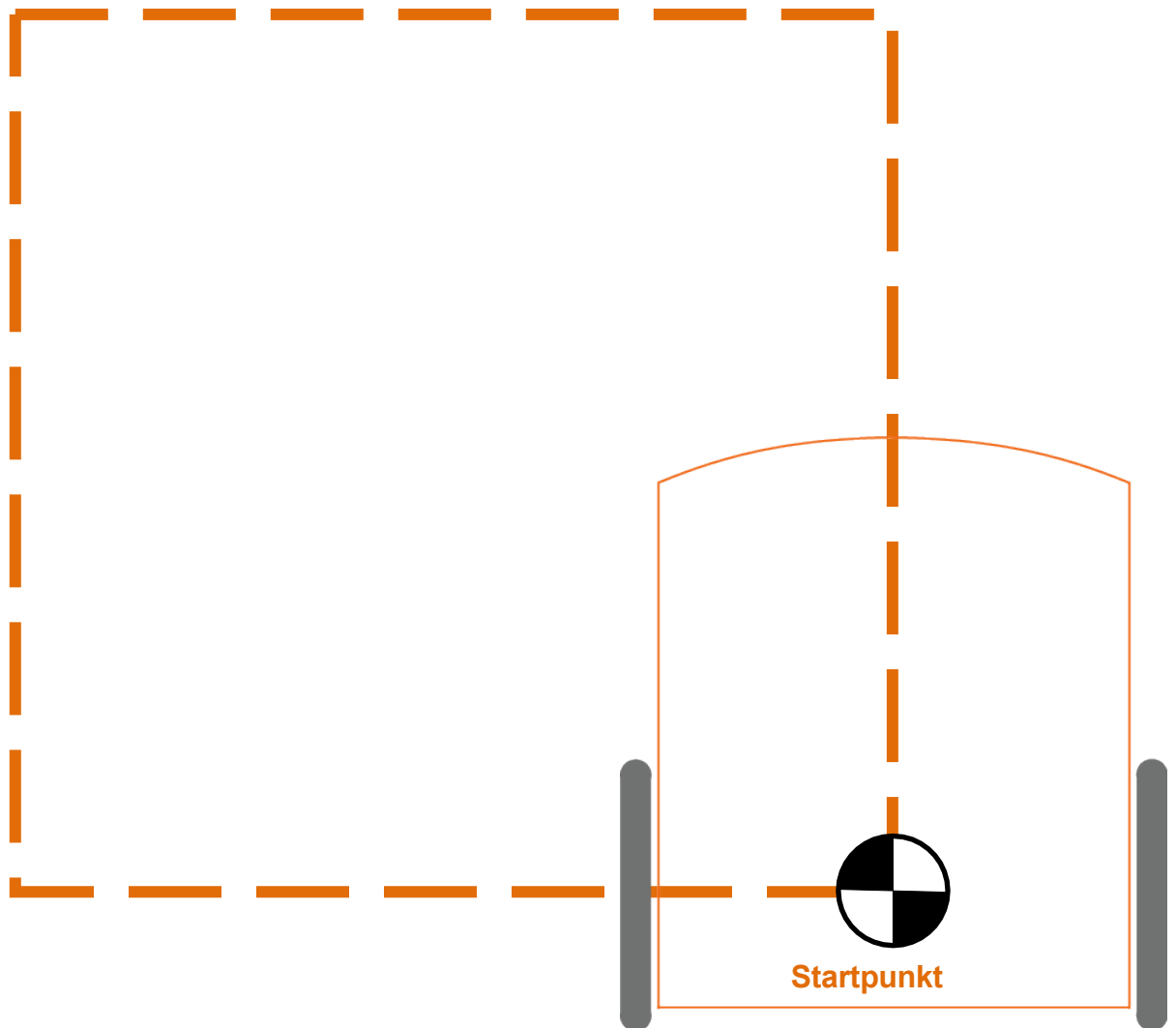
Name \_\_\_\_\_

### Optionale Herausforderung: Zeichne deine Form

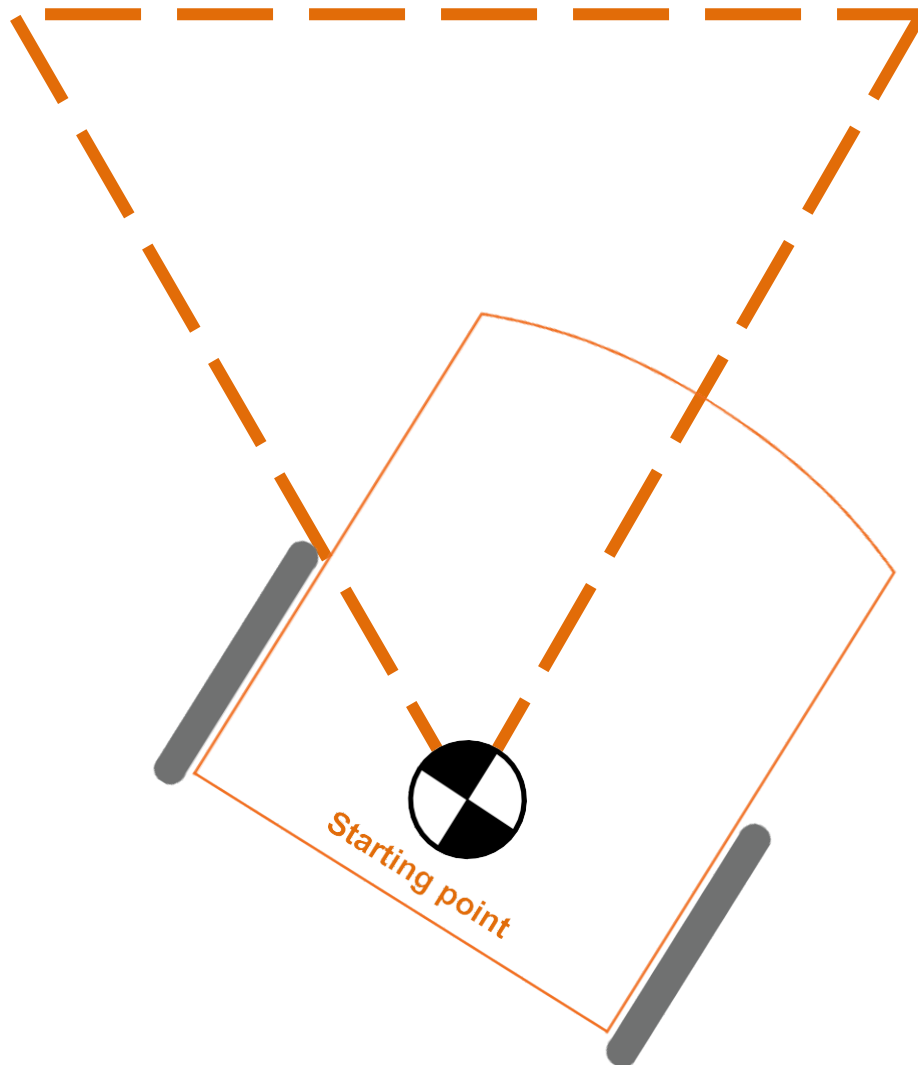
Befestige einen farbigen Marker an deinem Roboter. Stelle deinen Edison auf ein Blatt Papier und starte dein Kreisprogramm. Beobachte, wie der farbige Marker deine Form zeichnet, während sich der Roboter bewegt. Beobachte, ob dein Edison einen richtigen Kreis zeichnet oder nicht.

Name \_\_\_\_\_

## Lektion 4: Aktivitätsblatt 4.1

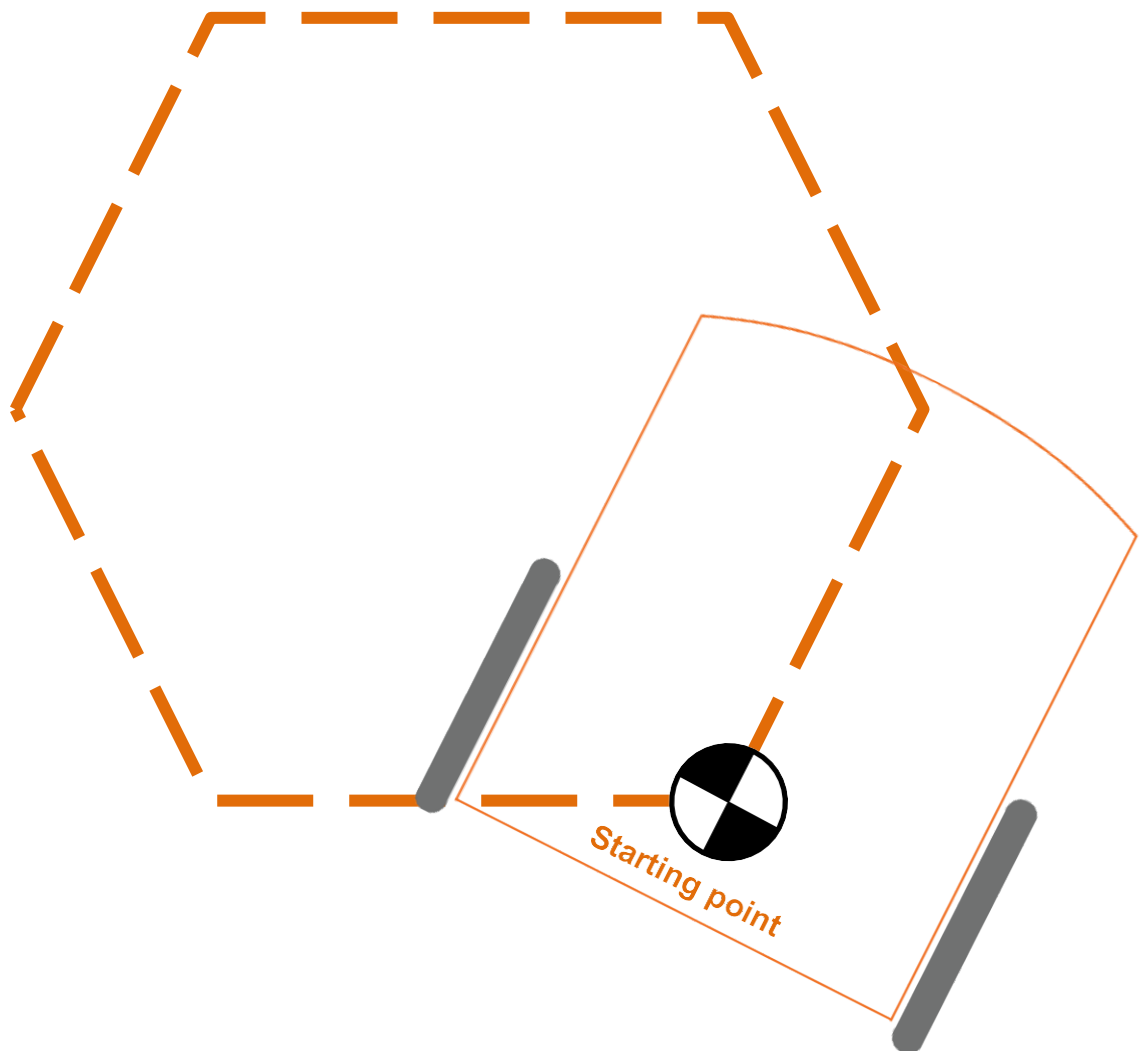


## Lektion 4: Aktivitätsblatt 4.2





## Lektion 4: Aktivitätsblatt 4.3



Name \_\_\_\_\_

## Lektion 4: Aktivitätsblatt 4.4

