



- Anlegen einer **For**-Schleife zur Erstellung des Polygons. Die darin enthaltenen Anweisungen werden **n**-mal ausgeführt.

Die folgenden Anweisungen befinden sich im Modul **ti\_rover**.

- Geradeausfahrt um **l** dm.
- Pause von 2s.
- Linksdrehung um den Winkel **a**.
- Pause von 2s.
- Abspeichern der Koordinaten der Eckpunkte in **xx** und **yy**.
- **xx** und **yy** in die Listen **x** und **y** exportieren.

**Lehertipp** : Die Anweisungen **rv.waypoint\_x()** und **rv.waypoint\_y()** findet man im Menü **Path** im Modul **ti\_rover**.

- Erstellen der Grafik wie bisher schon behandelt. Es ist auch eine grafische Darstellung außerhalb Pythons durch die Listen **x** und **y** möglich.

```

1.1 *Dok RAD 25/25
*U6APP.py
for i in range(n):
    rv.forward(l)
    sleep(2)
    rv.left(a)
    sleep(2)
    xx.append(rv.waypoint_x())
    yy.append(rv.waypoint_y())
store_list("x",xx)
store_list("y",yy)

```

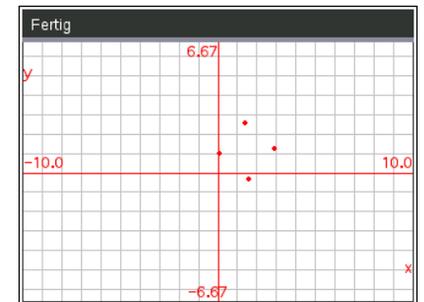
```

1.1 *Dok RAD 34/34
*U6APP.py
# Grafik
plt.cls()
plt.grid(1,1,"solid")
plt.axes("on")
plt.labels("x","y")
plt.color(255,0,0)
plt.scatter(xx,yy,"o")
plt.show_plot()
return

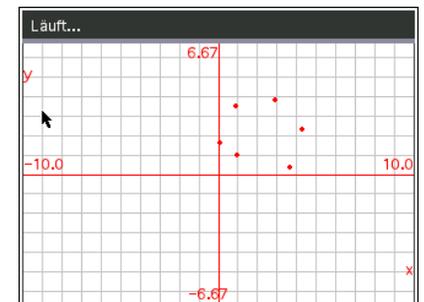
```

**Ergebnis :**

Startet man das Programm und ruft man über `var` die Funktion **poly( )** mit den Argumenten **poly(4, 2)** auf, so erhält man die Eckpunkte eines Quadrates von 2 dm Seitenlänge. Es sollte sich ein Bild ähnlich wie rechts abgebildet ergeben. Durch Reibung während des Drehens kann die Figur zudem leicht verzerrt aussehen.



So sieht die Grafik bei einem Hexagon aus, wenn die Drehungen nicht so gut ausgeführt sind.



Mit den in den Taschenrechner exportierten Listen lässt sich die nebenstehende Grafik erzeugen, deren Punkte man mit `trace` abtasten und die Koordinaten bestimmen kann.

**Lehertipp:** Für diese Art von Übung sollte man nicht die Anweisungen `rv.pathlist_x ()` und `rv.pathlist_y ()` verwenden. Tatsächlich zeichnet der Rechner beim Zeichnen einer Seite die Koordinaten der Eckpunkte der Seite des Polygons auf. Die Koordinaten des letzten Punktes sind also die Koordinaten des ersten Punktes der nächsten Seite. Mit den Anweisungen `rv.pathlist_x ()` und `rv.pathlist_y ()` erhält man die Endpunktkoordinaten also doppelt.

**Hinweis:** Das Format des Rasters sollte man abhängig von den Polygonen machen, die gezeichnet werden sollen. Hier wurden die Standardeinstellungen verwendet, um die Genauigkeit der ROVER-Spur zu beobachten.

Man sollte auch auf die Art der Oberfläche achten, auf der sich der Roboter bewegt. Diese sollte nicht zu viel Widerstand gegen Bewegung leisten oder im Gegenteil das Rutschen fördern. Es ist auch auf eine gute Beweglichkeit der Kugel zu achten – andernfalls ergibt sich halt ein Bild wie oben ...

