

## TGW Robo Challenge 2020 | Reglement (Stand: Juli 2019)

Es gibt drei Einzelbewerbe:

1. Logistics Race LEGO Klasse
2. Logistics Race offener Bewerb
3. Logistics Hackathon

Die Regeln für das Logistics Race sind so konzipiert, dass sie mit einem LEGO Mindstorms-Baukasten oder auch mit einem selbst gebauten Roboter realisiert werden können.

Für jeden einzelnen Bewerb gibt es eine getrennte eine Einzelwertung. Die Sieger bzw. die 2. und 3. Plätze werden bei den drei Bewerben jeweils mit einem Preis belohnt.

### **1. Logistics Race**

Die folgenden Regeln für die Logistics Race gelten 1:1 sowohl für die LEGO Klasse als auch für den offenen Bewerb mit selbstgebauten Robotern.

Ziel dieses Bewerbs ist es, eine Box und die darin enthaltene Ladung ins Ziel zu transportieren. Dort wird die Kiste wieder von einem Fördersystem und in ein vollautomatisiertes Hochregallager übernommen. Die Kiste darf bereits vor dem Start vom Roboter aufgenommen werden, es muss allerdings gewährleistet sein, dass die Kiste noch befüllt werden kann. D.h. die Kiste muss von oben frei zugänglich sein. Am Weg zum Ziel warten verschiedene Hindernisse auf die Roboter. So befindet sich bspw. ein „Mitarbeiter“ auf der die Linie. Es kann auch zu Unterbrechungen der Linie selbst kommen. Um den Weg ins Zentrallager so schnell wie möglich zu erreichen, gibt es eine farblich markierte Abkürzung durch das Zwischenlager. Jedoch muss hier besonders aufgepasst werden, da nur wenig Platz zwischen den Regalen vorhanden ist. Der schnellste Roboter gewinnt. Wird die Kiste bzw. Teile der Ladung während des Laufs verloren, gibt es eine Zeitstrafe. Die Zeitmessung wird beim Durchfahren eines Lichtschrankens am Start gestartet und beim Durchfahren am Ziel wieder gestoppt.

Eine mögliche Strecke ist in der unten dargestellten Abbildung skizziert.

- **Allgemeine Spezifikation**

Die Untergrundfarbe des Kurses ist weiß. Die Linie ist schwarz und 15 mm breit. Im Startbereich befindet sich eine Paketaufnahmestelle, von der die Box aufgenommen werden muss sowie ein Förderband am Ende, auf das das Paket wieder abgelegt werden muss.

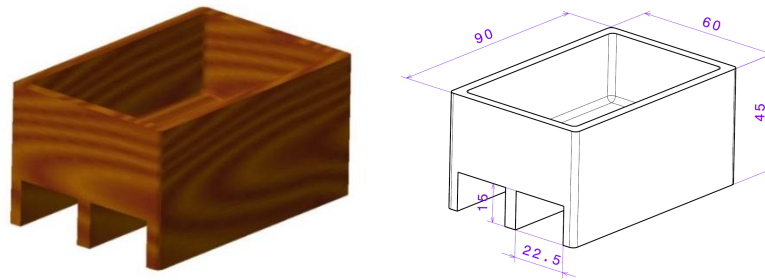
- **Roboterbetrieb**

Die Roboter müssen alle Aufgaben autonom erledigen. Jeder Eingriff durch Personen bzw. das Verlieren der Ladung oder der Kiste wird mit einer Zeitstrafe von 30 Sekunden geahndet.

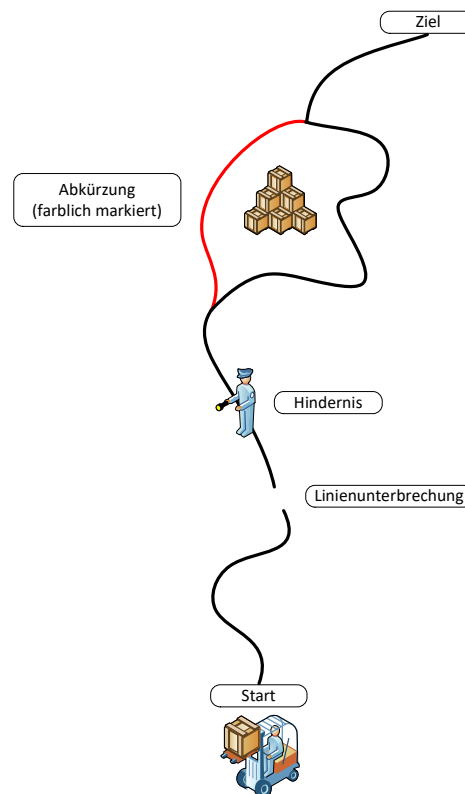
- **Zeitlimit**

Die Roboter haben zur Bewältigung der Aufgaben maximal 3 Minuten Zeit. Bei Überschreitung wird der Lauf abgebrochen und es erfolgt keine Wertung.

- **Verlust der Linie**  
Jeder Roboter, der den Linienverlauf verliert, muss den Kurs an der Stelle erneut aufnehmen, an dem der Linienverlauf erfolgte – alternativ an einem vorherigen Punkt (z. B. bereits überquerte Stelle).
- **Linienunterbrechung**  
Die Linie ist im Bereich des Laufes für maximal 10 cm unterbrochen.
- **Hindernis auf der Linie**  
Es befindet sich ein Hindernis mit den maximalen Abmessungen LxBxH 25x6,5x20 cm und einem minimalen Gewicht von 1 kg auf der Linie. Das Hindernis darf berührt, jedoch nicht umgefahren werden.
- **Abkürzung**  
Die Abkürzung ist farblich markiert. Bedeutet, der Linienabschnitt im Bereich der Abkürzung wechselt von Schwarz auf Rot. Weiters befinden sich im Bereich der Abkürzung auf beiden Seiten Regale. Dadurch verengt sich der Fahrweg auf 30 cm.
- **Größe des Pakets**  
Das Paket hat eine Größe von LxBxH: 90x60x45 mm.



Symbolbild für den Parcours:



## Bewerbe

- **Vorrunde**  
Für die Vorrunde wird eine Zeit von 3 Stunden anberaumt. In diesem Zeitfenster können so viele Läufe mit Zeitmessung erfolgen wie möglich. Es wird jedoch darauf geachtet, dass Teams mit weniger aufgezeichneten Läufen bevorzugt werden. Die schnellsten 16 Teams qualifizieren sich für das Finale.
- **Finalrunde**  
Es wird in gestürzter Reihenfolge der Qualifikation gestartet – d.h. das Team auf Rang 16 beginnt, und das Team mit der schnellsten Zeit aus dem Vorlauf startet zum Schluss. Jedes Team hat nur einen Lauf. Die Zeit aus der Vorrunde wird gelöscht.
- **Sieger**  
Sieger ist das Team, das in der Finalrunde die Bestzeit erreicht hat.

## 2. Logistics Hackathon

„Der Roboter startet an der Ladestation und muss von einem der Wareneingangsdocks die Ware abholen und ins Hochregallager bringen. Anschließend ist aus einem anderen Bereich des Hochregallagers eine Ware abzuholen und zu einem der Warenausgangsförderer zu bringen. Nach

getaner Arbeit soll der Roboter wieder zurück zur Ladestation fahren. Aber Achtung, der Weg kann durch Arbeiter oder Hindernisse blockiert sein.“

Ziel dieses Bewerbs ist es, dass ein Roboter einen vorgegebenen Parcours abfährt. Dabei folgt der Roboter keinem vorgegebenen Weg, sondern erhält an der aktuellen Position nur die Koordinate des nächsten Wegpunktes. Dieser Hinweis kann aus einem QR-Code (in der Nähe der angegebenen Koordinate) gelesen werden. Sobald der Roboter den QR-Code eingelesen hat, muss er diesen – mit einem Zeitstempel versehen – an den Leitreechner senden. Zwischen den einzelnen Punkten muss der Roboter zwischen den Hindernissen frei navigieren. Als Roboter-Plattform wird ein Turtlebot 3 verwendet.

**Achtung: während des Bewerbs wird keine Ware transportiert, sondern die Wegpunkte werden nur angefahren.**

- **Allgemeine Spezifikation**

Der Untergrundfarbe des Kurses ist weiß. Die Arena hat eine Größe von 3 x 3 m.

- **Roboterbetrieb**

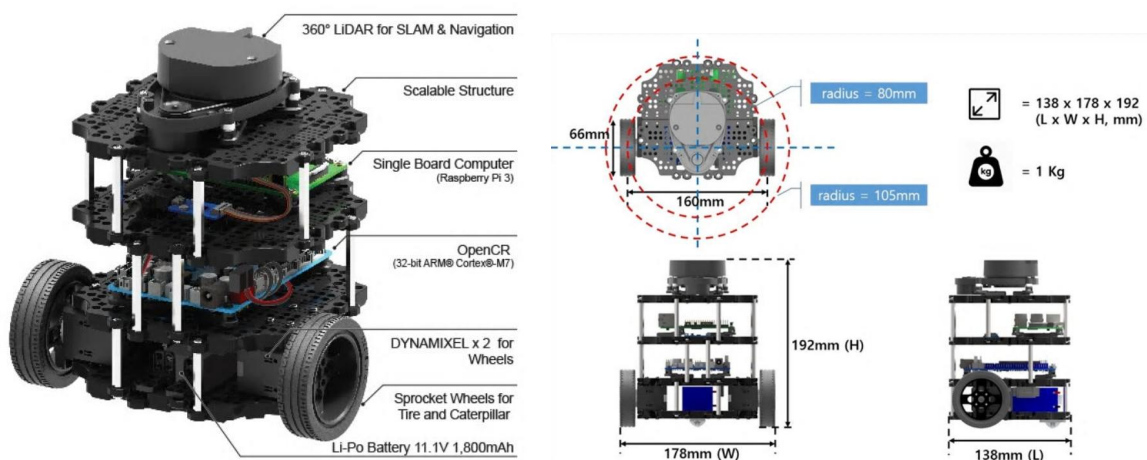
Die Roboter müssen alle Aufgaben autonom erledigen. Jeder Eingriff durch Personen wird mit einer Zeitstrafe von 30 Sekunden geahndet.

- **Zeitlimit**

Die Roboter haben maximal 5 Minuten zur Bewältigung der Aufgaben Zeit. Bei Überschreitung wird der Lauf abgebrochen, und der letzte „gefundene“ Wegpunkt mit dessen Scanzeit in die Wertung genommen.

- **Roboter**

Als Standard-Plattform wird ein Turtlebot 3 Burger verwendet (siehe Abbildung) – mehr Informationen unter <http://emanual.robotis.com/docs/en/platform/turtlebot3/specifications/>. Die Roboter werden vom Veranstalter vor Ort zur Verfügung gestellt. Als Betriebssystem steht ein Linux Ubuntu 16.04 und ROS Kinetic zur Verfügung. Alle erforderlichen Hardwaretreiber sind am Raspberry Pi vorinstalliert.



Der Roboter ist mit einem 2D-Laserscanner (RPLIDAR 1 – Messbereich: 0,12 – 3,5 m; 360°; Winkelauflösung 1°), einer Kamera (Raspberry Pi Camera Module V2 – Auflösung: 8 Megapixel) und einer 9DOF IMU ausgestattet. Als Recheneinheit wird ein Raspberry Pi 3 verwendet. Die Kamera ist in Fahrtrichtung nach vorne ausgerichtet.

- **Hindernisse**

Es befinden sich mehrere Hindernisse im Bereich der Arena. Diese werden vor den Läufen „zufällig“ im Parcours platziert. Das Hindernis darf berührt, jedoch nicht umgefahren werden. Die Hindernisse können in Größe, Form und Farbe variieren und sollen Einrichtungsgegenstände eines Logistikzentrums repräsentieren.

- **QR-Codes**

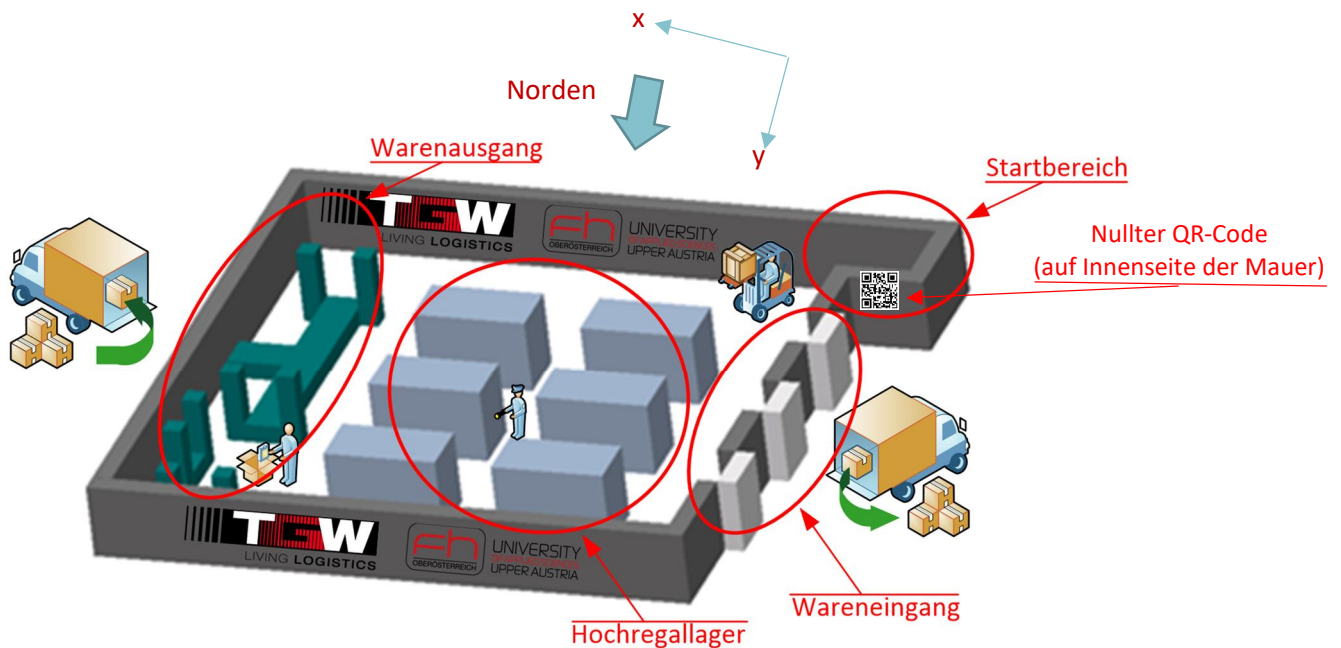
Zur Navigation sind mehrere (bis zu 6) QR-Codes in der Arena angebracht. Die QR-Codes liefern einen Hinweis auf den nächsten Wegpunkt. Die QR-Codes werden Online → <http://gogr.me/de/> generiert. Der QR-Code enthält eine Nummer, einen Text und die x- bzw. y-Koordinate als auch die Orientierung (als Himmelsrichtung) des QR-Codes am nächsten Wegpunkt. *Beispiel:* Der QR-Code in der folgenden Abbildung enthält den Hinweis: 1, Wareneingang, 1.4, 1.7, W (d.h. 1. Wegpunkt, Text: „Wareneingang“, nächster Wegpunkt: x-Koordinate 1,4 m, y-Koordinate 1,7 m, Orientierung Richtung Westen)



- **Start und Koordinatensystem**

Der Roboter startet auf dem Koordinatensystemursprung (0,0) und ist dabei nach Norden orientiert. Die x-Achse des Koordinatensystems zeigt nach Osten, die y-Achse nach Norden. Der nullte QR-Code befindet sich zum Startzeitpunkt direkt im Sichtfeld des Roboters. Der nullte QR-Code ist somit nach Süden orientiert. Er enthält die Koordinaten für den ersten Wegpunkt.

In der folgenden Abbildung ist die Umgebung des Bewerbs schematisch dargestellt.



- **Zeitmessung**

Zur Zeitmessung muss der Roboter jedes Mal, wenn er einen QR-Code liest, eine ros-Message vom Typ `tgw_qr_reading` auf dem Topic `/tgw_qr_reading` publishen. Das entsprechende `.msg` File kann auf der Homepage der RoboChallenge ([www.tgw-group.com/robochallenge](http://www.tgw-group.com/robochallenge)) heruntergeladen werden. Während der Challenge wird sich ein Leitrechner auf den roscore des Roboters verbinden und die Nachrichten auf dem Topic `/tgw_qr_reading` zur Zeitmessung benutzen. Der Zeitpunkt, zu dem die jeweilige Message am Leitrechner ankommt, gilt je als Zeitpunkt des Erreichens des Wegpunkts.

## Bewerb

Der Bewerb ist in zwei Abschnitte eingeteilt:

- Den Teilnehmern werden eine mögliche Umgebung und der Roboter als Simulationsmodell (ROS/Gazebo) zur Verfügung gestellt. Dadurch können ohne Kostenaufwand die Aufgabenstellungen (QR-Code-Reader, Navigationsmodul, Mapping, ...) in der Simulation entwickelt werden.
- Am Wettbewerbstag wird jeder Gruppe ein Roboter zur Verfügung gestellt und die Teams haben 6 Stunden Zeit, um ihren Code aus der Simulation am realen Roboter und der realen Umgebung zu erproben.
  - a. Um 15 Uhr werden die Roboter und deren Leitreechner bei der Bewerbsleitung abgegeben und das Finale beginnt. Somit kann gewährleistet werden, dass kein Team benachteiligt wird. Hier werden die anzufahrenden Wegpunkte und Hindernisse neu verteilt.
  - b. Sieger ist das Team, das alle Wegpunkte in der kürzesten Zeit abgefahren hat. Zur Kontrolle muss der Roboter die eingelesenen QR-Codes – mit einem Zeitstempel versehen – zum Leitreechner schicken. Diese Zeit wird für die Zeitrechnung herangezogen. Sollten nicht alle Wegpunkte erreicht werden, gewinnt das Team, das am weitesten gekommen ist. Haben mehrere Teams den gleichen Wegpunkt erreicht, wird das Team vorgereiht, das den Wegpunkt zu einem früheren Zeitpunkt erreicht hat.