

mBot-RETTUNGSFAHRZEUG

Die Stadt hat ein Problem: In einem engen Labyrinth steckt ein kleines „Opfer“ (z. B. ein Lego-Figur).

Der mBot soll als autonomes Rettungsfahrzeug programmiert werden, das:

1. eigenständig einem Linienpfad folgt,
2. Hindernisse erkennt,
3. und am Ziel automatisch stoppt, um zu „retten“.

Arbeite die folgenden 6 Schritte der Reihe nach durch, um das Rettungsfahrzeug erfolgreich zu programmieren.



Checkliste und Ablaufplan:

Aufgaben:	erledigt:
<p>1. Was ist ein mBot? (Motoren, Sensoren, Controller)</p> <p>Sensoren erkannt (Linienfolger & Ultraschallsensor) mBot eingeschaltet mBot mit App/Computer verbunden Testprogramm ausgeführt (z. B. Sensorwert anzeigen)</p> <p style="text-align: right;">Schritt 1</p>	
<p>2. Den Linienfolgersensor entdecken!</p> <p>Sensor unter dem mBot gefunden Sensorwert bei hell und dunkel getestet Werte in die Tabelle eingetragen</p> <p style="text-align: right;">Schritt 2</p>	
<p>3. Test auf dem schwarzen Balken</p> <p>Schwarzen Balken auf Papier gezeichnet mBot neben/auf dem Balken getestet</p> <p style="text-align: right;">Schritt 3</p>	
<p>4. Auf der Linie anhalten</p> <p>Programm erstellt: „solange Sensor \neq Stop-Wert \rightarrow fahren“ Stoppen bei hell ausprobiert Passenden Stop-Wert gefunden und eingetragen</p> <p style="text-align: right;">Schritt 4</p>	

5. Der mBot folgt der Linie!

Sensorwerte für „links runter / rechts runter“ bestimmt
Korrekturprogramm gebaut (kurz nach rechts/links drehen)
mBot auf der Linie getestet
Verhalten korrigiert (Geschwindigkeit/Drehzeit angepasst)

Schritt 5**6. Das Opfer ist gefunden!**

mit Ultraschallsensor verschiedene Entfernungen gemessen
Einen sinnvollen Stop-Abstand gewählt
Programm hinzugefügt: „Wenn Abstand < ... → Stop“
mBot erfolgreich am Opfer gestoppt

Schritt 6