

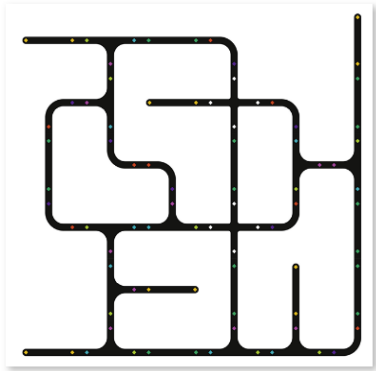


Okay, der eXperiBot hat jetzt einen Plan „B“ und eine Fehlerkontrolle, die dafür sorgen, dass er nicht in einem Rundgang hängen bleibt, wenn er ungünstig startet. Gelangt er trotzdem in einen Rundgang, kreist er erst einige Male, bevor die Fehlerkorrektur die Situation erkennt und den Fehler meldet.

So richtig professionell ist das noch nicht! Immerhin ist der eXperiBot eine Maschine, die über ein sehr gutes Gedächtnis verfügt.

Es gab da mal vor vielen Jahren einen 12-jährigen Jungen aus England, der hat das Gedächtnis seines Roboters geschickt genutzt und ein Stück Orientierung in den Algorithmus gebracht.

Vielleicht kannst du etwas von dieser Idee für den eXperiBot übernehmen?



Spielplan B

**Programmieraufgabe**

1. Der Algorithmus soll wie folgt funktionieren. (Nutze die Befehle aus der Befehlsbox.)

- ➔ Beende die Fahrt, wenn du das Ziel erreicht hast.
- ➔ Fahre geradeaus, solange du nicht abbiegen musst.
- ➔ Wenn du abbiegen musst:
  - Wenn du nach rechts abbiegen musst, werde zum Linksabbieger.
  - Wenn du nach links abbiegen musst, werde zum Rechtsabbieger.
  - Zähle in beiden Fällen Rechts-Drehungen mit +1, Links-Drehungen mit -1 (auch die erste Drehung!).
  - Ist die Summe aller Drehungen null, fahre wieder geradeaus bis du das nächste Mal abbiegen musst.

**Befehlsbox**

erhöhe Drehung um -1

wiederhole bis Richtung = [Dropdown] [Dropdown]

mache [Dropdown]

2. Der folgende Algorithmus ist unvollständig. Bitte erweitere ihn so, dass er wie oben beschrieben funktioniert. Teste und verbessere den Algorithmus ausgiebig.

Starte eXperiBot

setze Drehung auf 0

wiederhole bis Richtung = [Dropdown] [Dropdown]

mache

+ falls Drehung = 0 und Weg nach [Dropdown]

mache Fahre Fahrtrichtung [Dropdown]

sonst falls [Dropdown]

mache [Dropdown]

um rechts fahren

Fahre Fahrtrichtung [Dropdown]

erhöhe Drehung um 1

um links fahren

Fahre Fahrtrichtung [Dropdown]

erhöhe Drehung um -1



Teste, ob man mit deinem Algorithmus von jedem Startpunkt aus ein Ziel erreichen kann.