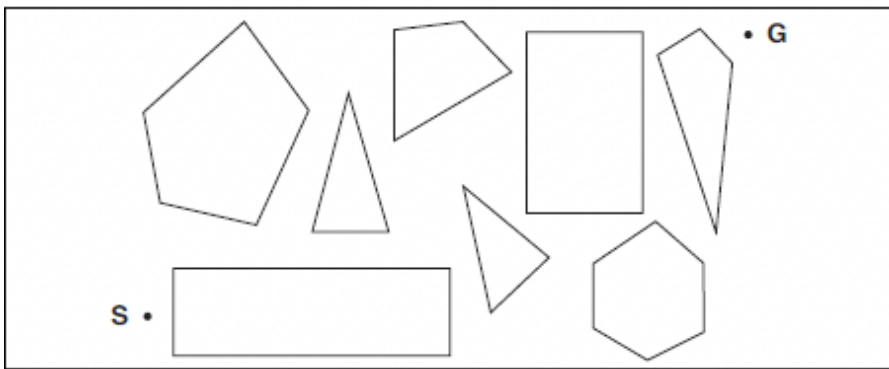


### 3. Labor zur

## Künstliche Intelligenz

- 1) Schreiben Sie Python Code (in Google Colab wenn möglich) für Labor2 - Übung G11. Ihr sollt 2 verschiedene Algorithme benutzen: best-first-Suche (Greedy) und A\*.
- 2) Gegeben sei das Problem den kürzesten Weg zwischen zwei Punkte, S und G in der Ebene zu finden. Zusätzlich dazu gibt es in dieser Ebene mehrere polygonale Hindernisse (s. Bild) und es gilt diese umzugehen. Dieses Beispiel ist eine Vereinfachung der Situation in der ein Roboter in einem überfüllten Raum navigieren muss.



- a) Erklären Sie weshalb der kürzeste Weg von einem Polygonknoten zu einem anderen aus geradlinigen Strecken zwischen verschiedene Polygonknoten bestehen muss. Definieren Sie einen geeigneten Zustandsraum. Wie groß ist jetzt der Zustandsraum?
- b) Schreiben Sie Python Code und benutzen Sie das A\* Algorithmus, um dieses Problem zu lösen. Der Roboter verdient 1000 Punkte, wenn er den Ziel erreicht, und verliert 1 Punkt für jedes mm, das er fährt. Wählen Sie die Größe der Ebene, sodass es sinnvoll ist. Nachdem der Roboter den Ziel erreicht, wird er teleportiert irgendwo zufällig in der Ebene und muss er wieder den Ziel erreichen. Das wiederholt sich 100 mal.
- c) Verändern Sie die Umgebung, so dass in 30 Prozent der Fälle ihr Agent ein ungewünschtes Ziel erreicht. Dieses Ziel wird zufällig aus der Menge der sichtbaren Kanten gewählt. Falls keine sichtbaren Kanten existieren, dann bewegt sich der Agent überhaupt nicht. Dies ist ein Modell der Bewegungsfehlern eines realen Roboters. Ändern Sie ihren Agenten, so dass bei der Entdeckung eines Fehlers, er sofort seine Position berechnet und ein Plan erstellt, um zu seiner früheren Position zurückzukehren und von dort weiter, in der richtigen Richtung, zu machen. Vorsicht, manchmal kann auch diese Aktion fehlerhaft sein! Geben Sie ein Beispiel eines Agenten, der zweimal hintereinander einen Bewegungsfehler macht und trotzdem sein Ziel erreicht.
- d) Jetzt kann der Roboter nur eine Strecke  $d$  in allen Richtungen sehen. Modellieren Sie das Problem. Was passiert, wenn  $d$  sehr klein ist?

Hinweis: Für beide probleme, soll man graphisch sehen, wie die Algorithme funktionieren.