

Praktikum 4: Weitere Suchverfahren & Heuristiken

In diesem Praktikum wird die Klasse *Suche* erweitert, um verschiedene Suchstrategien durchführen zu können. Dafür müssen die Klassen *Suche* und *Knoten* wieder angepasst werden.

Aufgabe P4

Implementiert die aus der Vorlesung bekannten Suchstrategien: Breitensuche, UCS, Greedy-Search und A-Stern.

1. Erweitert die Klasse *Suche* um die notwendigen Attribute und Methoden für die gewünschten Suchstrategien.
2. Erweitert die action-Methode des *MyAgent* so, dass ihr schnell zwischen den möglichen Suchstrategien umschalten könnt.
3. Gebt nach jedem Suchdurchlauf jeweils die Größe der open- und closedlist auf der Konsole aus.
4. Testet und protokolliert eure Suchverfahren in den folgenden Welten. Versucht vorher zu schätzen, wie groß die Anzahl an Knoten in open- und closedList wohl sein wird.
(Hinweis: Abhängig von eurer Implementierung kann z.B. A* auf der Karte PW_easy_01 wenige Sekunden benötigen, unter Umständen aber auch 1-2 Minuten)

Welt: PW_L2	openList	closedList	Länge der gefundenen Lösung (in Aktionen)
Tiefensuche			
Breitensuche			
Greedy-Search			
UCS			
A-Stern			

Welt: PW_easy_01	openList	closedList	Länge der gefundenen Lösung (in Aktionen)
Tiefensuche			
Breitensuche			
Greedy-Search			
UCS			
A-Stern			

Welt: PW_easy_02	openList	closedList	Länge der gefundenen Lösung (in Aktionen)
Tiefensuche			
Breitensuche			
Greedy-Search			
UCS			
A-Stern			

Zusatzaufgabe (für Interessierte):

Implementiert man die Knoten und die Suche recht naiv, stellen schon moderate Pac-Man Welten eine Herausforderung an Laufzeit und Speicher dar. Bspw. lässt sich die **optimale Lösung** für die „PW_FHDO“-Welt (siehe Abbildung 1) ohne Optimierungen auf den meisten Rechnern nicht ermitteln. Der dafür benötigte Arbeitsspeicher kann locker 16 GB übersteigen. Verändert man die Welt nur leicht, (z.B. je einen horizontalen Tunnel durch die Buchstaben D und O) steigt der benötigte Speicher auf über 100 GB!

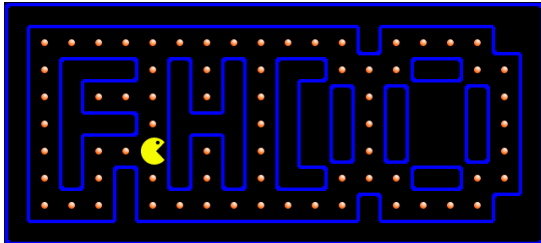


Abbildung 1: Die FH-DO Pac-Man Welt.

1. Verbessert die Klasse Knoten hinsichtlich des benötigten Speichers. Hier ein paar mögliche Anregungen:

- Welche Informationen sind für das Ziel alle Dots zu fressen relevant und verändern sich im Laufe des Spiels? Müssen solche konstanten Informationen in jedem Knoten gespeichert werden?
- Ist es sinnvoller, wenn ein Knoten seine Kindknoten oder seinen Vorgänger kennt?
- Muss das 2D Array mit den Positionen der Dots für jeden neuen Kindknoten kopiert werden oder nur in bestimmten Fällen?

Mögliche Optimierungen des Laufzeit- und Speicherverhaltens des gesamten Suchvorgangs betreffen nicht nur die Knoten, sondern auch die Klasse Suche. Besonders das Laufzeitverhalten wird stark durch die Suche beeinflusst, da regelmäßig große Listen von Knoten durchlaufen werden müssen.

2. Verbessert die Klasse Suche hinsichtlich der Laufzeit und des benötigten Speichers. Hier ein paar mögliche Anregungen:

- Welche Datenstrukturen sind am effizientesten für die open- und closedList?
Für die openList sind die folgenden Gegebenheiten zu berücksichtigen:
 - Es werden häufig Knoten hinzugefügt bzw. entfernt.
 - Für die heuristischen Suchverfahren muss die openList sortiert gehalten werden, bzw. es muss häufig die ganze openList durchsucht werden, um die richtige Stelle zum Einfügen zu finden.Für die closedList gilt:
 - Es werden häufig Knoten hinzugefügt.
 - Sehr häufig muss geprüft werden, ob sich ein Knoten schon in der closedList befindet.
 - Die closedList speichert alle bereits expandierten Knoten komplett, d.h. auch Informationen die eigentlich zur Überprüfung, ob ein Knoten schon in der closedList ist, nicht notwendig sind.
- Muss jede simulierte Aktion des Pac-Man einen eigenen Knoten erzeugen oder können mehrere Aktionen zu einem Knoten zusammengefasst werden? Bzw. können große

Teilbäume des Suchbaums zusammengefasst werden und in einem Schritt durchsucht werden?

3. Testet euren Agenten auf den folgenden Welten mit einem Suchverfahren eurer Wahl. Es kann sein, dass sich einige Welten nicht in realistischer Zeit lösen lassen. Evtl. ist es nötig, dass Zeitlimit für eine Aktion in den Optionen des Servers anzupassen.

Testet und protokolliert eure Suchverfahren in den folgenden Welten:

Welt	openList (geschätzt)	closedList (geschätzt)	openList (real)	closedList (real)	Länge der gefundenen Lösung (in Aktionen)
PW_FHDO					
PW_advanced_01					
PW_advanced_02					