Hilfestellung

Wir haben ein Grid mit mehreren Nodes.
Jeder Node hat 3 Kosten.
Das waagrechte und senkrechte überqueren einer Node hat die Kosten von 10.
Das diagonale überqueren einer Node hat die Kosten von 14.

- G Cost = Kosten vom Startpunkt.
- H Cost = Kosten um den Endpunkt zu erreichen.
- F Cost = Die Gesamtkosten also G+H

Ermittelt werden diese Kosten indem man eine startNode und endNode hat. Wichtig ist hierbei auch currentNode, also der Node auf dem sich der Agent gerade befindet.
Wir haben eine Klasse Pathfinding dort werden die Kosten der Nachbar Nodes des currentNodes verglichen und die Node mit den geringsten Kosten wird abgelaufen. Dieser Prozess wiederholt sich bis EndNode erreicht wurde.

Um auf cognitve maps zu erweitern gibt es 2 Ideen.

Idee 1:
Wir haben mehrere Grids die wiederum logischerweise mehrere Nodes haben, dadurch können wir Räume besser aufteilen. Jedes Grid kann noch die Eigenschaft haben was für ein Raum er ist (Geschlossen, Flur).
Beim Betreten einer bestimmten Node kann die Sichtbarkeit eines anderen Grids getriggert werden, heißt ich als Agent laufe auf eine Node und kenne somit einen neuen Raum. Wie man das mit A* kombiniert stellt sich mir schwierig dar, da wir dann Zwischenziele bräuchten. Außerdem gibt es das Problem der checkNeighbour-Methode. Da jede Grid Klasse eine Funktion hat um die Nachbarn zu überprüfen und abzulaufen, heißt jedes Grid hat seinen eigenen Index. Wenn zwei Grid zusammengefügt sind, ist es schwierig zwischen beiden Grids die Kosten zu überprüfen.

Idee 2:
Wir bleiben bei einem Grid und geben manuell bestimmten Nodes extrem hohe Kosten, damit der Agent nicht direkt zum Ausgang findet. Er wird dann wahrlos umher laufen, deswegen müssen wir darauf überprüfen, ob ein Nachbar diese extrem hohe Anzahl an Kosten hat, daraufhin geben wir ihm wieder die Kosten die er haben sollte. Dadurch haben wir quasi einen A*-Algorithmus der zwischenzeitig stoppt und die Route wieder neu sucht.
Problematisch hier ist, wie gibt man einzelnen Nodes bestimmte Kosten.
Ansatz: Programm startet erst beim Klicken auf ein UI-Element. Vorher können wir durch Mausklicks einzelnen Nodes hohe Kosten geben um somit Räume zu markieren.
Problem: Der Raum müsste eigentlich komplett gefüllt werden mit hohen Kosten, daraufhin wäre er aber wieder komplett bereinigt mit seinen eigentlichen Kosten, also wie machen wir nur einen bestimmten Teil begehbar, anstatt wieder alles. Das erinnert mich leicht an MineSweeper.