

Dokumentation Aufgabe 2: Ameisenfutter

Lösungsidee

Das Areal ist in Felder eingeteilt, wobei jedem Feld eine X-Koordinate und eine Y-Koordinate zugeteilt. Diese Felder werden als Erstes entsprechend der Arealgröße erzeugt. Dabei hat das Feld in der linken oberen Ecke die Koordinaten (0|0). Danach wird das Nest auf bestimmten Felder erzeugt und die Futterquellen auf zufälligen Feldern verteilt. Nun wird das Areal in der Konsole ausgegeben.

Nun beginnt die eigentliche Simulation. Bei jedem Feld wird nacheinander überprüft ob eine Ameise ohne Futter auf ihm steht, wenn ja bewegt sich die Ameise. Dafür werden die benachbarten Felder(nicht diagonal) nach Duftpunkten und Futterquellen untersucht. Steht die Ameise neben einer Futterquelle betritt sie diese, sonst betritt sie das Feld mit den meisten Duftpunkten. Sollte es mehrere Felder mit den meisten Duftpunkten geben läuft die Ameise zufällig auf eines dieser Felder.

Danach wird das Areal auf Ameisen, die bereits Futter tragen überprüft. Jene laufen einfach auf direktem Weg zurück zum Nest und hinterlassen auf dem Feld auf dem sie stehen einen Duftpunkt.

Die Ameisen die auf Futterquellen stehen werden als nächstes bewegt. Sie verlassen die Futterquelle Richtung Nest und ziehen der Futterquelle einen Futterpunkt ab.

Schließlich sind die Ameisen, die sich noch im Nest befinden, dran. Sie verlassen das Nest nach einem bestimmten Muster. Erst am linken oberen Feld des Nests, als Zweites am rechten Oberen, dann am linken Unteren und zuletzt am rechten Unteren.

Jeder dieser Aktionen sorgt für ein erneutes Ausgeben des Areals.

Nach einer bestimmten Anzahl von Durchläufen wird das Verschwinden des Duftes simuliert.

Jedem Feld wird ein Duftpunkt abgezogen, wobei der Wert nicht unter 0 fallen kann.

Zuletzt wird noch überprüft ob die Ameisen das gesamte Futter des Areals in ihr Nest gebracht haben, sollte dies so sein wird das Programm beendet.

Umsetzung im Programm

Die Lösungsidee wurde in Java umgesetzt. Das Programm besteht aus zwei Klassen: Die Klasse *Map* stellt Variablen für das Areal und die Anzahl von Ameisen und Futtereinheiten im Nest, sowie alle Methoden zur Erzeugung des Areals, der Generierung der Futterquellen und des Nests, der Bewegung der Ameisen, der Ausgabe des Areals und zum Verlieren des Duftes zur Verfügung. Die Klasse *Controller* enthält die Variablen, die veränderbar sein sollen und die Grundvoraussetzungen der Simulation bestimmen (z.B. Arealgröße, Anzahl der Ameisen, ...), erstellt jeweils ein Objekt der Klassen und regelt den Ablauf der Simulation.

Als Erstes wird ein Objekt der Klasse *Controller*, danach ein Objekt der Klasse *Map* mit einem Konstruktor erzeugt, so wird das *Controller*-Objekt übergeben und die Variablen gleich mit den richtigen Werten gefüllt. Der zweite Schritt ist das Füllen der beiden Arrays, deren Indizes den Koordinaten entsprechen. Der *field*-Array wird mit leeren Feldern und der *foodPointsAndOlfactoryPoints*-Array mit Nullen, da bis zu diesem Zeitpunkt nur ein leeres Areal vorhanden ist. Dies ändert sich mit dem nächsten Schritt: Das Nest und die Futterquellen werden generiert, das Nest abhängig von den Koordinaten der rechten

unteren Ecke, die als Variablen angegeben sind, im *field*-Array gespeichert. Die Futterquellen hingegen werden zufällig auf noch freien Feldern generiert.

Nachdem die Grundvoraussetzungen des Areal nun vorhanden sind beginnt die eigentliche Simulation, deren Schritte in einer for-Schleife, die scheinbar unendlich ist und nur dadurch beendet wird, dass alle Futtereinheiten im Nest sind, was eine if-Bedingung in der Schleife überprüft, indem sie überprüft ob die Anzahl der Futtereinheiten im Nest den Futtereinheiten pro Futterquelle multipliziert mit der Anzahl der Futterquellen, da das die Anzahl aller Futtereinheiten ist, entspricht, stehen. Die Simulation unterscheidet zwischen Ameisen, die am Rand, in einer Ecke oder in der Mitte stehen, weil diese unterschiedlich viele angrenzende Felder und damit Bewegungsmöglichkeiten haben. Ansonsten läuft die Simulation wie in der Lösungsidee beschrieben ab. Nach jeder Aktion wird das Areal neu abgedruckt. So werden die Ameisen bewegt bis sie alles Futter in ihr Nest gebracht haben.

Beispiele

Beispiele werden in gesonderten Textdateien beigelegt.