Software Engineering Praxis (WS 2004/2005)

Abgabetermin: 12. November 2004

1 Friss oder stirb! (10 Punkte)

Der Planet Wa-Tor hat die Form eines Torus und ist vollständig mit Wasser bedeckt. Im Wasser von Wa-Tor leben zwei Tierarten: Haie und Fische. Die Haie ernähren sich von Fischen. Die Nahrung der Fische ist unbekannt, aber offenbar immer reichlich vorhanden. Der Planet Wa-Tor ist in quadratische Felder eingeteilt, die zusammengefasst als Rechteck dargestellt werden können. In jedem Feld befindet sich entweder ein Hai, ein Fisch oder er ist unbesetzt.

Nachbarfelder sind Felder, die vertikal oder horizontal aneinandergrenzen (also nicht diagonal). Wa-Tor ist ein riesiger WAsser-TORus, d. h. er hat die Form eines gigantischen Ringes. Rollt man den Ring auf, so erhält man ein Rechteck. Es gelten dann zyklische Randbedingungen: geht man von der obersten Zeile einen Schritt nach oben so landet man in der untersten Zeile, analoges gilt bei den Spalten.

Jede Zeiteinheit besteht aus zwei Phasen:

Fische schwimmen und vermehren sich:
Jeder Fisch wählt sich zufällig eines der freien Nachbarfelder aus und zieht dorthin. Erreicht der Fisch ein bestimmtes Alter (fbrut), so entsteht an der alten Position ein neuer Fisch. Beide Fische haben dann das Alter 0. Gibt es kein freies Nachbarfeld, so findet auch keine Vermehrung statt.

Haie jagen und vermehren sich:
Jeder Hai sucht aus den Nachbarfeldern, auf denen Fische sind, zufällig eines aus und zieht dorthin, wobei der Fisch aufgefressen wird. Ansonsten wählt er zufällig ein freies Nachbarfeld aus. Wenn ein Hai fasten Zeiteinheiten lang keinen Fisch gefressen hat, stirbt er. Bei Erreichen des Alters hbrut entsteht an der alten Position ein neuer Hai.

Schreiben Sie ein Java-Programm zur Simulation des Ökosystems auf dem Planeten Wa-Tor. Achten Sie dabei auf die sinnvolle Verwendung objektorientierter Konzepte. Der Anfangszustand sollte zufällig bestimmt werden, gesteuert durch Parameter (Feldgröße, fbrut, hbrut und fasten), die als Konstanten gegeben sind. Die Simulation sollte dann solange erfolgen, bis ein Abbruch durch Ctrl-C erfolgt. Jeder Schritt soll natürlich ausgegeben werden. Sie können die Klasse *Escape* verwenden, um die Ausgabe schön zu gestalten. Stellen Sie den Zustand jeweils als Karte dar (in Form eines Rechtecks): „*“ für Haie, „.“ für Fische und „ “ für freie Felder). Nach jedem Simulationsschritt sollte 1 Sekunde gewartet werden.

Zu Wa-Tor gibt es einen ausführlichen Artikel in einem Sonderheft des *Spektrum der Wissenschaft: Computer-Kurzweil*.

Vernünftige Parameter sind für 32 * 14 Felder: 200 Fische und 20 Haie zu Beginn, fbrut = 3, hbrut = 10, fasten = 3.

Tipp: Eine mögliche Aufteilung in Klassen könnte wie folgt aussehen:

- *Wator*:
  Repräsentiert den Planeten und enthält ein zweidimensionales Array mit Feld-Objekten. Methoden sind vorhanden zum initialisieren des Planeten, zur Ausgabe des Zustandes und zur Simulation (mit einer Endlosschleife).
- *Feld*:
  Ein Feld hat möglicherweise ein Tier als Bewohner und ein Array von Nachbarfeldern (die in Wator gesetzt werden). Von außen kann man den Bewohner ansehen und ändern, die Nachbarn setzen (das ist für Wator wichtig) und mit je einer Methode einen unbewohnten Nachbarn bzw. einen „Fisch-Nachbarn“ zufällig auswählen.
- *Tier*:
  Ein Tier hat ein Alter und weiss, auf welchem Feld es sich momentan befindet. Das Alter kann man lesen, erhöhen und auf 0 setzen. Außerdem kann es prinzipiell ziehen, wobei das erst die konkreten Tiere entscheiden wie das passiert.
- *Fisch* und *Hai*:
  Beide sind Unterklassen von Tier und legen im Prinzip nur noch fest, wie das ziehen, d. h. ein Simulationsschritt funktioniert.

Beachte: Da sich die Position von Tieren verändern kann, muss man darauf achten, dass ein und dasselbe Tier im selben Schritt nicht zweimal gezogen wird!
Viel Erfolg!