Dr. Andreas Borchert
Ingo Melzer

Sektion Angewandte Informationsverarbeitung

26. Juli 1999
Ferienblatt

Allgemeine Informatik II (SS 1999)

Abgabetermin Ende Oktober 1999

11 It's Just a Game

Nach dem großen Erfolg des Wettbewerbs zum fünften Übungsblatt haben wir beschlossen, daß etwas vergleichbares über die Semesterferien stattfinden sollte. Um den Spaß etwas zu erhöhen, haben wir uns für ein Spiel entschieden. Es wird ein Server geschrieben werden, so daß verschiedene Programme auf diesem Weg gegeneinander spielen können.

Schon jetzt einmal die Regeln des Spiels; gespielt wird auf einem sieben auf sieben Felder großem Schachbrett. Ein Spieler startet mit einem Stein rechts unten in der Ecke und einem links oben. Der andere Spieler startet mit je einem Stein in den beinden verbleibenden Ecken. Nun ziehen die Spieler abwechselnd bis das ganze Brett gefüllt ist. Bei jedem Zug muß ein Spieler, falls noch möglich, mit genau einem Stein ziehen. Dabei hat er zwei Möglichkeiten:

- Er kann auf ein freies, benachbartes Feld ziehen, wobei sein alter Stein erhalten bleibt (man gewinnt also einen Stein) oder
- er springt in ein zwei Felder entferntes, freies Feld. Hierbei wird sein Stein verschoben.

In beiden Fällen sind diagonale Züge zulässig. Im ersten Fall gibt es also bis zu acht mögliche Ziele, im zweiten bis zu sechszehn. Nach einem Zug werden alle Steine des Gegners, die mit dem neuen Stein benachbart sind, mit der Farbe des aktiven Spielers gefärbt.

Die Frage die sich jetzt natürlich noch stellt ist: wie schreibe ich meinen Clienten zu diesem Spiel. Dies ist eigentlich nicht sonderlich schwer. Es reicht fast aus, wenn man eine Prozedur MakeMove schreibt, die automatisch von einem elektronischen Spielleiter aufgerufen wird, wenn man einen Zug abgeben soll. Beim Aufruf bekommt man die aktuelle Spielsituation und die Zeit, die man für den Rest der Partie noch hat, übergeben. Als Antwort wird im *VAR-Parameter* ein legaler Zug erwartet. Damit die Verständigung mit dem Rest der Welt auch einwandfrei funktioniert, sollen zwei Module implementiert werden. Das erste ist sehr schnell erstellt, indem man in Greedy.om einfach alle vorkommen von *GreedyPlayers* durch den Namen des eigenen Moduls mit der Speilstrategie ersetzt. Dementsprechend kann als Grundlage für das zweite Modul einfach GreedyPlayers.om genutzt werden. Man sollte als erstes in Zeile 54 den Namen des eigenen Moduls eintragen. Ab jetzt kann an der eigenen Strategie (in MakeMove) gebastelt werden. Beachtet bitte schon jetzt, daß es eine Zeitbeschränkung geben wird. Vermutlich wird jeder Spieler drei Minuten für eine ganze Partie bekommen. Die aktuelle Restzeit bekommt man immer beim Aufruf von MakeMove) mit übergeben. Beim Erzeugen des *Makefiles* sollte der Parameter `+l puzzle` nicht vergessen werden. Also z. B. `mmo -c makefile +l puzzle +l terminals`

Zusätzlich kann es ganz lehrreich sein, einen Blick auf die Implementierung der eingebundenen Module zu werfen. Ausführbare Module zum Testen, inklusive *matches* zum Austragen von Partien, liegen auf der Thales unter `/usr/local/games/puzzle/`.