Prof. Dr. Franz Schweiggert Abteilung Angewandte Informationsverarbeitung 22. Januar 2002
Christian Ehrhardt Blatt 11

Allgemeine Informatik 3 (WS 2001/2002)

Abgabetermin 29.01.2002

1 MUD - Multi User Dungeon

Aufgabe dieses Blattes ist es, ein kleines Programm zu schreiben, mit dem man sich durch die verschiedenen Räume eines Irrgartens bewegen kann. In jedem Raum kann man zusätzlich noch Gegenstände aufnehmen oder ablegen. Der Irrgarten selbst wird durch spezielle Dateien und Verzeichnisse vorgegeben:

- Jeder Raum wird durch ein Verzeichnis beschrieben. Der Name des Verzeichnisses ist gleichzeitig eine Kurzbeschreibung für den Raum.
- Jede reguläre Datei in einem solchen Verzeichnis beschreibt entweder einen Gegenstand oder einen Ausgang, der in einen anderen Raum führt. Reguläre Dateien mit Länge 0 sind Gegenstände, der Name der Datei ist gleichzeitig der Name des Gegenstandes. Reguläre Dateien mit einer Länge größer als 0 sind Ausgänge, der Name der Datei beschreibt den Ausgang (z.B. "Holztuer"), der Inhalt der Datei gibt an in welchen Raum der Ausgang führt (z.B. "Kueche"). Hinter dem Namen des Raums kann möglicherweise noch Whitespace stehen, der nicht zum Namen gehört.
  Alle nicht regulären Dateien sollen ignoriert werden. Das gilt insbesondere auch für symbolische Links.
- Alle Verzeichnisse, die die Räume beschreiben sind Unterverzeichnisse des aktuellen Verzeichnisses.
- Beim Programmstart beginnt die Reise immer in einem fest vorgegebenen Raum (z.B. "Lobby").
- Ein Abenteurer, der sich durch den Irrgarten bewegt kann keine zwei Gegenstände mit dem selben Namen bei sich tragen. Auch in einem Raum können keine zwei Gegenstände oder Ausgänge mit dem selben Namen existieren.

Euer Programm soll folgende Befehle von der Standardeingabe einlesen und verarbeiten können:

look Den aktuellen Raum anschauen. Folgende Informationen sollen dazu ausgegeben werden:

- Der Name des Raums.

- Eine Liste aller Gegenstände, die sich im Raum befinden.
- Eine Liste aller Ausgänge und wohin sie führen.

go *exit* Einen Ausgang benutzen. Dabei muß der Name des Ausgangs und nicht der Name des Zielraums angegeben werden. Im Beispiel oben wäre also “go Holztuer” möglich, “go Kueche” dagegen nicht.

inventory Zeigt alle Gegenstände an, die der Abenteurer gerade bei sich trägt.

take *object* Einen Gegenstand aufnehmen. Dadurch soll er natürlich aus dem Raum verschwinden und bei inventory auftauchen. Nur Gegenstände können aufgenommen werden, keine Ausgänge oder gar ganze Räume.

drop *object* Einen Gegenstand den man bei sich trägt ablegen. Das ist das Gegenstück zu take.

Ein Spieler sollte die Gegenstände, die er bei sich trägt *NICHT* verlieren, wenn das Programm beendet wird, auch dann nicht, wenn das Programm durch einen kill Befehl beendet wird. Um das zu erreichen sollten die aufgenommenen Gegenstände in einem privaten Verzeichnis abgelegt werden. Dieses private Verzeichnis kann sich irgendwo im Heimatkatalog befinden.
Bitte beachtet auch, daß sich möglicherweise mehrere Eurer Programme gleichzeitig durch den Irrgarten bewegen können. Es ist also damit zu rechnen, daß Gegenstände in einem Raum zu jeder Zeit auftauchen oder verschwinden können.
Achtung: Um Gegenstände aufzunehmen oder abzulegen müssen Dateien gelöscht werden. Achtet dabei bitte darauf, daß nicht versehentlich wichtige Daten gelöscht werden. Für diese Übungsaufgabe müssen nur Dateien der Länge 0 gelöscht werden. Es empfiehlt sich, das vor dem eigentlichen Löschen zu testen.
Hinweis: Einen Beispielirrgarten gibt es auf der Homepage der Vorlesung als .tar.gz Archiv. Die Datei kann mit dem Kommando “tar -xvzf mud.tar.gz” entpackt werden, die eigentlichen Daten befinden sich dann im Unterverzeichnis “mud”.
Hinweis: Bei diesem Blatt sollen soweit wie möglich direkt die Unix-Systemaufrufe (read, write, open, unlink etc.) verwendet werden. Die Funktion fopen ist für dieses Blatt nicht erlaubt!