Dr. M. Grabert
Johannes Mayer

Abteilung Angewandte Informationsverarbeitung

2. Juni 2001
Blatt 6

Systemnahe Software (SS 2001)

Abgabetermin: 18. Juni 2001

Die Lösung für eine der beiden folgenden Aufgaben müssen Sie schon bis zum 11. Juni 2001 abgeben. Welche Lösung Sie früher abgeben bleibt Ihnen überlassen.

9 Wir basteln mit „Rohren“! (10 Punkte)

Nehmen Sie zuerst Ihren persönlichen Vorlesungsbegleiter zur Hand und schlagen diesen auf den Seiten 67/68 auf. Dort finden Sie eine Erklärung der Bibliotheksbefehle `popen()` und `pclose()`. Sie würden nun, nachdem Sie die Beschreibung gelesen haben, gerne wissen, wie diese implementiert sind. Na das kriegen Sie doch raus! Sie programmieren `popen()` und `pclose()` einfach selbst! ;-) Übertreiben müssen wir es aber dann doch nicht, oder?! Also gut, nach einem `popen()` muss zunächst das zugehörige `pclose()` kommen, bevor `popen()` erneut aufgerufen wird. Das macht die Sache jetzt aber wirklich leichter!

Tipps: Mit `execl("/bin/sh", "sh", "-c", cmd, NULL);` können Sie das Kommando `cmd` von der Shell ausführen lassen (unter Berücksichtigung von PATH). Mit `fdopen()` kann aus einem File Descriptor ein File Pointer gewonnen werden. Den File Descriptor darf man dann aber nicht mehr schließen, sondern muss am Ende den File Pointer schließen!

10 Wer wird Millionär? (10 Punkte)

Sie haben doch sicherlich in letzter Zeit 'mal in die „Glotze“ geschaut? Dann haben Sie bestimmt auch auf die Sendung „Wer wir Millionär?“ gezappt? Na das wäre doch die Idee, dieses Spiel zu implementieren. Damit Sie nicht ganz im Regen stehen, können sie das *ausführbare Programm* der Beispiellösung auch schon vorab testen. Noch nicht zufrieden? Klar, eine *Datenbank* mit Fragen bekommen Sie natürlich auch. Jetzt aber ans Werk!

Spielregeln: Der Kandidat bekommt vom Showmaster nacheinander Fragen gestellt bis er eine Frage falsch beantwortet oder Millionär ist. (Mit Jokern befassen wir uns erst gar nicht; das ist doch Kinderkram!) Zu jeder Frage werden dem Kandidaten vier Antworten zur Wahl gestellt, von denen er eine auswählen kann. Wählt er keine aus und beendet das Spiel, so hat er das bisher erspielte Geld gewonnen. Wählt er die falsche Antwort aus, so hat er verloren, das Spiel ist beendet und er erhält 32000 DM bzw. 1000 DM, falls er mindestens so viel zuvor erspielt hatte, und andernfalls geht er leer aus, wenn er zuvor weniger als 1000 DM erspielt hatte. Tippt er die richtige Antwort, so kommt er in die nächste Gewinnstufe – außer er hat schon eine Million erspielt; dann ist das Spiel zu Ende und er hat die Million gewonnen – und bekommt die nächste Frage gestellt. Der Kandidat beginnt zunächst mit 0 DM. Danach kann er in den folgenden Schritten bis zu einer Million „klettern“: 100 DM, 200 DM, 300 DM, 500 DM, 1000 DM, 2000 DM, 4000 DM, 8000 DM, 16000 DM, 32000 DM, 64000 DM, 125000 DM, 250000 DM, 500000 DM und 1000000 DM.

Aufbau der Datenbank: Die Datenbank ist ein Datei, die für jede Frage genau sechs Zeilen enthält. Davon enthält jeweils die erste die Frage, die nächsten vier enthalten die Antworten und in der letzten steht die Nummer der richtigen Antwort (zwischen 1 und 4).

Aufgabenstellung: Schreiben Sie eine Client-Server-Anwendung, wobei der Client aus dem Server durch ein `fork()` entsteht. Beide kommunizieren miteinander über zwei Pipes miteinander (in jede Richtung eine Pipe).

Der Server (also der Vater-Prozess) liest zunächst die Datenbank mit den Fragen komplett ein. Danach wählt er zufällig eine dieser Fragen aus und vertauscht die Antworten dieser Frage auch zufällig (und vergisst dabei nicht, die Nummer der Lösung anzupassen). Dann schickt er die Frage mit den Antworten (ohne die Lösung) an den Clienten zusammen mit dem Geldwert (d. h. auf wieviel Geld er durch die richtige Beantwortung dieser Frage kommen kann) dieser Frage. Dann bekommt er vom Clienten den Tipp und antwortet ihm nach der Auswertung des Tipps. In der Antwort teilt er dem Clienten mit, ob das Spiel zu Ende ist oder weiter geht. Ist es zu Ende, dann erfährt der Client durch diese Antwort auch den gewonnenen Geldbetrag und im Falle einer falschen Antwort die richtige Lösung. Andernfalls erfährt er den aktuell erspielten Geldbetrag. Geht das Spiel weiter, so stellt der Server nun die nächste Frage. Der Server achtet natürlich darauf, dass bei einem Spiel keine Frage doppelt gestellt wird!

Der Client liest zunächst vom Server die Frage mit Antworten und den Geldwert der Frage. Nachdem er diese ausgegeben hat, fordert er den Spieler auf, eine Auswahl zu treffen. Diese Auswahl ist entweder eine Antwort oder Spielende. Diese Wahl teil er dem Server mit und erhählt von diesem die Antwort (siehe oben). Daraufhin beendet er sich oder liest vom Server die nächste Frage, je nach dem was in der Antwort des Servers stand. Natürlich macht er für den Spieler entsprechende Ausgaben, damit dieser mitkriegt wie ihm gerade geschieht.

Tipps: Die *Fragen-Datenbank* muss sich im selben Katalog befinden wie das *Demo-Programm*. Werfen Sie doch 'mal einen Blick auf die Beispiel-Programme. Weitere Fragen gibt es unter *http://www.rtl.de/*. Sie können die Fragen-Datenbank nach Lust und Laune erweitern! :-))

Viel Glück mit dem Millionen!