

Blatt 3

Client-Server

Aufgabe 1: Parallel-Server

In der Vorlesung wurde die Client-Server-Kommunikation mit Sockets und `fork()` erklärt. Schreibt ein Client-Programm und ein Server-Programm.

a) Client

Das Client-Programm soll eine Socket-Verbindung mit dem Server-Programm herstellen und anschließend alle Tastatureingaben direkt an den Server weiterleiten. Zum Beenden bricht man einfach mit `Strng-C` ab.

Das Client-Programm soll zwei Aufruf-Parameter haben, und zwar den Namen des Hosts, zu dem eine Verbindung hergestellt werden soll, und die Nummer des Ports dort.

b) Server

Das Server-Programm soll an einem Port auf eine eingehende Verbindung von einem Client-Programm warten. Sobald ein Client eine Verbindung hergestellt hat, soll sich der laufende Server-Prozeß mithilfe von `fork()` in zwei Prozesse aufspalten. Der erste Prozeß soll erneut auf eine weitere eingehende Verbindung warten. Der zweite Prozeß soll die eingegangene Verbindung bearbeiten. Er soll alle Daten, die darüber ankommen, solange in eine Datei schreiben, bis die Verbindung wieder abgebaut wird. Die Dateien sollen `daten1.txt`, `daten2.txt` usw. heißen, die Datei jedes Kind-Prozesses soll also eine eigene Nummer haben. Außer in die Dateien soll der Server die Daten auch auf die Standardausgabe schreiben, jeweils mit dem Namen der Ausgabedatei davor. Zum Beenden bricht man ebenfalls einfach mit `Strng-C` ab. (Dies beendet auch automatisch alle Kind-Prozesse, zumindest unter Linux.)

Das Server-Programm soll bei seinem Aufruf einen Parameter haben, und zwar die Nummer des Ports, an dem der Server auf eingehende Verbindungen warten soll.

Die Programme sollen wie immer in C geschrieben werden. Um die Programme auszuprobieren, ist es am besten, wenn man auf dem Linux-Rechner mehrere X-Terminals mit je einer Textkonsole öffnet. In einem Fenster kann man dann den Server starten, in den anderen jeweils einen Client.

Der Server muß selbstverständlich als erstes gestartet werden. Sollte es dabei eine Fehlermeldung geben, daß der Port schon belegt ist, muß man einen anderen nehmen. Schließlich arbeiten evtl. andere Studenten gleichzeitig an derselben Aufgabe. Die Ports mit Nummern kleiner als 1024 sind übrigens vom System belegt und können von normalen Benutzern nicht zum Warten auf eine Verbindung verwendet werden.