

## Aufgabe 5:

### trsh (12 Punkte)

Programmieren Sie (in zweier Gruppen) basierend auf der mysh von Aufgabe 4 eine um Hintergrundprozesse, Signalbehandlung und Zeitabrechnung erweiterte Shell: **trsh** (**t**ime **r**ecording **s**hell). Sie können dafür die Lösung (`~i4sp/pub/aufgabe4/mysh.c`), welche ab 04.12.2003 verfügbar ist verwenden. Die unten gestellten Fragen sind in der Dokumentation (**trsh.txt**) zu erläutern.

**a) Promptsymbol**

Ändern Sie das Promptsymbol der Shell von "mysh>" auf "trsh>" und passen Sie das Makefile an die neue Aufgabe an.

**b) Hintergrundprozesse**

Wenn eine Kommandozeile mit dem Zeichen `&` abgeschlossen wird, soll das Kommando wie bei einer UNIX-Shell als *Hintergrundprozess* ausgeführt werden. Die Shell soll nicht auf das Terminieren dieses Hintergrundprozesses warten, sondern sofort wieder das Promptsymbol ausgeben und das nächste Kommando von der Standardeingabe einlesen.

**c) Warten auf Kindprozesse**

Was passiert, wenn Hintergrundprozesse und ein Vordergrundprozess laufen und ein Hintergrundprozess zuerst fertig wird (☞ Dokumentation)? Stellen Sie in Ihrem Programm sicher, dass garantiert immer der Exit-Status des Vordergrundprozesses ausgegeben wird. Zum Testen können Sie das Programm **sleep** als Hintergrundprozess bzw. Vordergrundprozess verwenden.

**d) Signalhandler**

Der shell-Prozess soll nun das Interrupt-Signal vom Terminal abfangen. In der Signalbehandlungsfunktion soll zuerst einmal nur die Meldung "Interrupt!" auf dem Standardfehlerkanal ausgegeben werden. (**sigaction(2)**) Was passiert, wenn Ihr shell-Programm ein Interrupt-Signal erhält und nur ein Vordergrundprozess läuft bzw. wenn auch Hintergrundprozesse laufen? (☞ Dokumentation) Ändern Sie das Programm nun so, dass die Hintergrundprozesse das Signal SIGINT ignorieren. Was hat sich dadurch am Verhalten bei einem SIGINT an die shell geändert? (☞ Dokumentation)

**e) Versenden von Signalen**

Der trsh-Prozess soll nun zusätzlich den Sohnprozessen ein SIGQUIT schicken, wenn er selbst ein SIGINT erhalten hat (**kill(2)**).

**f) Jobverwaltung**

Implementieren Sie in der Shell ein Kommando **jobs**, das die Kommandozeilen und Prozeß-Ids aller laufenden Hintergrundprozesse ausgibt. Stellen Sie sicher, daß Ihre Jobliste immer aktuelle Informationen enthält, d.h. wenn ein Hintergrundprozeß terminiert, soll er auch aus der Jobliste ausgetragen werden. Geben Sie beim Terminieren eines Hintergrundprozesses auch dessen Exitstatus aus. Zur Verwaltung der Hintergrundprozesse (Jobs) verwenden Sie bitte die Implementierung aus `~i4sp/pub/aufgabe5/joblist.[ch]`. (Makefile anpassen!)

**g) Zeiterfassung**

Erweitern Sie die Shell nun so, dass jeweils unmittelbar nach dem Terminieren eines Sohnprozesses ein Abrechnungsdatensatz in die Datei `trsh.log` geschrieben wird. Der Abrechnungsdatensatz soll folgende Informationen über den terminierten Prozess enthalten (**times(2)**):

PID Exit-Status User-Time System-Time Kommandozeile

Die Zeiten sollen im Format *Sekunden:Millisekunden* ausgegeben werden.

**Abgabe: bis spätestens Donnerstag, 08.01.2004, 13:30 Uhr**