	Fachbereich 2
	Studiengang: <i>Ing.Informatik</i> Dozent: Döben-Henisch
	Prozessrechner – Übungsaufgabe Nr.5
	Letzte Änderung: 2.Juni 2004 Seite 1//2

Implementierung einer Interprozesskommunikation mit FIFOs

Beschreibung


Ein typischer Fall, bei dem mehr als ein Prozess auf eine bestimmte Ressource zugreift, ist der Datenaustausch über gemeinsame Speicherbereiche. Dafür gibt es mehrere Techniken (Pipe' s, Fifo' s, Shared Memory, Semaphoren...). In dieser Übung soll das Problem am Beispiel eines FIFO untersucht werden.

Ein FIFO wird auch als ' named pipe' bezeichnet. Es ist eine Datenstruktur, die wie eine Datei behandelt werden kann, ist aber in den Mechanismus einer Interprozesskommunikation eingebunden.

In der Übung Nr.5 soll jetzt eine FIFO-Struktur dazu benutzt werden, um die Kommunikation zwischen 3 Prozessen zu ermöglichen: ein Prozess A generiert Daten und zwei Prozesse B1 und B2 wollen diese Daten benutzen. Prozess A schreibt seine Daten in ein FIFO und die Prozesse B1 und B2 lesen die Daten aus dem FIFO. Unter Linux kann man diese Aufgabe z.B. dadurch lösen, dass man gleichzeitig 3 Shellfenster auf hat: in Shell 1 startet man Prozess A, in Shell 2 und 3 jeweils die Prozesse B1 und B2. Der Prozess A kann z.B. Daten vom Benutzer entgegennehmen und diese in das FIFO schreiben. B1 und B2 lesen jeweils das FIFO und zeigen den Inhalt am Bildschirm an.

Teilaufgaben

1. (MAX. PKT.: 2) Lesen Sie mit dem Shellbefehl ' man die Informationen zu den Begriffen ' fifo' ' mkfifo' , ' fopen' 'close' ; fread' , ' fwrite' , ' fclose' , ' lseek' ; feof' und ' perror' Mit ' man a BERGRIFF' koennen Sie alle Eintraege zu BERGRIFF lesen; mit ' man N BERGRIFF' lesen sie den N-ten Eintrag. Schreiben Sie eine kurze Erklärung, was ein FIFO ist und wie es funktioniert.
2. (MAX. PKT.: 3) Schreiben Sie drei kleine Testprogramme in C (oder C++), so dass ein Programm A Daten vom Benutzer holt und in ein FIFO schreiben kann. Zwei weitere Programme B1 und B2 holen diese Daten aus dem FIFO und zeigen sie dem Benutzer an.
3. (MAX. PKT.: 3) Welche Fehler könnten bei ihren Programmen auftreten? Wie können Sie ihre Programme so absichern, dass diese Fehler ausgeschlossen werden? Realisieren Sie diese ' sicheren' Programme.
4. (MAX. PKT.: 2) Unter <http://www.doeben-henisch.de/fh/II-PRT/II-PRT-TH/VL2/ii-prt-vl-vl2.html#Zeitmessen> wurde gezeigt, wie man unter Linux Zeit mit der Genauigkeit von Mikrosekunden messen kann. Benutzen Sie diese Möglichkeit, um zu messen, wie lange ein Programm B2 auf B1 warten muss (' Blocking Time'),um seine Daten lesen zu können.

	Fachbereich 2 Studiengang: <i>Ing.Informatik</i> Dozent: Döben-Henisch
	Prozessrechner – Übungsaufgabe Nr.5
	Letzte Änderung: 2.Juni 2004 Seite 2//2

Abgabe

Die Übungsaufgabe gilt als erfüllt, wenn folgende Bedingungen erfüllt sind:

1. In der Übung liegen alle Dokumente ausgedruckt vor (vergewissern sie sich rechtzeitig, wie und wo Sie ausdrucken können)
2. Sie können ihren Beitrag über den Zielrechner im PC-Labor per Beamer vorführen (Bedenken sie, dass unterschiedliche Präsentationsprogrammen gibt).
3. Alle an der Erstellung der Übung Beteiligten sind anwesend und erläutern den Beitrag (Wer nicht anwesend ist bekommt keine Punkte).