Universität Ulm - Abteilung Angewandte Informationsverarbeitung
Dr. Johannes Mayer und Hans Braxmeier, Sommersemester 2004

2. Übungsblatt zur Vorlesung Systemnahe Software II

Abgabetermin: Dienstag, 04.05.2004

Prozesse erzeugen mit fork() *(10 Punkte)*

Aufgabe 1: Einfacher Primzahltest (4 Punkte)

Schreiben Sie ein C-Programm, das testet, ob eine Zahl z, die über die Kommandozeile eingegeben wird, eine Primzahl ist oder nicht.

Dabei soll geprüft werden, ob z von einer Zahl im Intervall von 2 bis $\sqrt{z}$ geteilt wird. Teilen Sie dieses Intervall in weitere Bereiche auf und erzeugen sie für jeden dieser Bereiche einen neuen Prozeß. Findet ein Prozeß einen Teiler, so ist z keine Primzahl.

Achten Sie darauf, daß die Bereiche nicht allzu groß werden, d.h. Ihr Programm sollte nicht mehr als 20 neue Prozesse erzeugen.

Aufgabe 2: Matrizen (6 Punkte)

Schrieben Sie ein Programm, das als Eingabe zwei quadratische $n \times n$-Matrizen erhält und diese in n^2 Kindprozessen miteinander multipliziert ($n \leq 5$). Jeder Kindprozess berechnet ein Element der Ergebnismatrix und liefert den Wert als Exit-Status an den Erzeugerprozess, welcher die Teilergebnisse zusammenfügt und die Ergebnismatrix ausgibt. Es kann davon ausgegangen werden, dass alle Teilergebnisse (d. h. Elemente der Ergebnismatrix) im Bereich $\{0, \ldots, 255\}$ enthalten sind.

Tipp: Damit Sie wissen, wohin welches Ergebnis gehört, müssen Sie sich merken, welcher Kindprozess (PID!) welches Matrixelement berechnet. Dann können Sie aus der PID, die *wait()* liefert, darauf schließen, wohin das Teilergebnis (in der Ergebnismatrix) gehört.

Viel Erfolg!