Übungen zu Vorlesung
Betriebssysteme I
Wintersemester 1997/98

AG BS
Bettina Buth/ Jan Peleska

Serie 1

Prozeßverwaltung

Aufgabe 1: Petersons Algorithmus

Petersons Algorithmus zur Realisierung von *mutual exclusion* ist im Folgenden in ANSI C angegeben; er besteht aus 2 Prozeduren, die von den Prozessen aufgerufen werden, die auf die kritische Region zugreifen wollen.

```
#include ‘‘prototypes.h’’

#define FALSE  0
#define TRUE   1
#define N      2                /* number of processes                   */

int turn;                       /* whose turn is it?                     */
int interested[N];              /* all values initially 0 (FALSE)        */

void enter_region (int process) /* process: who is entering (0 or 1)     */
{
  int other;                    /* number of other process               */

  other = 1 - process;          /* the opposite of process               */
  interested[process] = TRUE;   /* show that you are interested          */
  turn = process;               /* set flag                              */
  while (turn == process && interested[other] == TRUE)  /* null statement  */
}

void leave_region (int process) /* process: who is leaving (0 or 1)      */
{
   interested[process] = FALSE; /* indicate departure from critical region */
}
```

a) Geben Sie (in FDR-Syntax) eine CSP-Spezifikation dieses Algorithmus' an, indem Sie zwei Prozesse spezifizieren, die möglichst direkte Umsetzungen der beiden Prozeduren `enter_region` und `leave_region` aufrufen.

b) Weisen Sie mit Hilfe von FDR nach, daß der Algorithmus auch tatsächlich *mutual exclusion* garantiert. Definieren Sie dazu eine geeignete Abstraktion des erwarteten Verhaltens. Dokumentieren Sie das Ergebnis geeignet.

Aufgabe 2: Verallgemeinerung auf $N \geq 2$ Prozesse

a) Die Verallgemeinerung von Petersons Algorithmus mit einem Array `interested[N]` für $N = 3$ ist keine Lösung für *mutual exclusion*. Begründen Sie diese Aussage.

b) Finden Sie eine korrekte Verallgemeinerung von Peterson's Algorithmus auf mehr als 2 Prozesse. Geben Sie dafür wieder eine CSP-Spezifikation an, und überprüfen Sie sie mit Hilfe von FDR.

Aufgabe 3

Welche Probleme bei der Prozeßverwaltung werden durch Petersons Algorithmus nicht gelöst? Wie könnte man sie im Rahmen des Algorithmus lösen ohne das Konzept grundsätzlich zu verändern?

Abgabe: bis Mo, 17.11.96, in der Übung