Christian-Albrechts-Universität zu Kiel
Institut für Informatik und Praktische Mathematik

Prof. Dr. W.-P. de Roever
Dipl. Inf. K. Baukus · Dr. M. Steffen

Informatik III

Wintersemester 2000/01

Serie 9

15. Januar 2001

Thema : Produzenten und Konsumenten

Ausgabetermin: 15. Januar 2001

Abgabe: 22. Januar 2001, bis 12:00 Uhr im Schrein

Wir haben in der Vorlesung anhand eines einfachen „Teletype"-Terminaltreibers gesehen, welch wichtige Rolle das Produzenten-Konsumenten-Problem in Betriebssystemen spielt. Man kann es abstrakt als einen grundlegenden Kommunikationsmechanismus betrachten, und zwar als asynchrone, also gepufferte, Kommunikation zur Datenübertragung vom Produzent zum Konsument.

Aufgabe 1: [Produzent/Konsument]

Die erste Aufgabe gehe, in einigermaßen unrealistischer Weise, von einem *unbeschränkten* Puffer aus. Der Puffer zur Entkoppelung von Produzenten und Konsumenten sei als array `Buff` vereinbart. Vervollständigen sie folgendes Codefragment.

```
/* producer/consumer pseudocode                          */

producer ()
{
  while (TRUE) {                    /* Produzent                */

    datum   = produce_item ();      /* Produzent tut seine Aufgabe */
    Buf[in] = datum;                /* Einfuegen                */
    in      = in + 1;               /* und weiterz"ahlen        */

  }
}

consumer ()                         /* Konsument                */
{
  while (TRUE) {

    datum = Buf[out];
    out   = out + 1;
```



```
    consume(datum);
  }
}
```

Aufgabe 2: [Endlicher Puffer]

Spezialisieren Sie die Lösung aus Aufgabe 1 zu einem *endlichen* Puffer der festen Größe n. Wie verhält sich Ihre Lösung, wenn man nicht nur einen Produzenten und einen Konsumenten, sondern mehrere erlauben möchte?

Unter *Verklemmung* versteht man intuitiv das unerwünschte Phänomen, wenn ein Prozeß nicht fortschreiten kann. Im Zusammenhang mit der Modellierung des Produzenten-Konsumenten-Szenarios bedeutet dies, daß der Produzent kein neues Datum abliefern kann, obwohl er er eines vorrätig hat, oder der Konsument keines lesen kann, obwohl ungelesene Daten im Puffer vorhanden sind. Anders ausgedrückt: daß die beiden Prozesse zu arbeiten aufhören, weil der Produzent schlicht nichts mehr produziert, ist *keine* Verklemmung. Frage: kann sich Ihre Lösung verklemmen?

Aushungern (*starvation*) ist ein weiteres unerwünschtes Prozeßverhalten. Intuitiv läßt es sich als der Umstand beschreiben, daß ein Prozeß zwar im Prinzip eine Aufgabe ausführen kann, aber ständig *übergangen* wird, beispielsweise, ein Prozeß ist zwar ständig im Zustand *ready*, es werden beim Kontextwechsel aber immer andere Prozesse ausgewählt. In Zusammenhang mit der Aufgabe interessiert uns: können sich Produzent und Konsument gegenseitig aushungern? Wie sieht es aus, falls man mehrere Produzenten (oder Konsumenten) hat?

Aufgabe 3: [Produzenten/Konsumenten-Problem mit binären Semaphoren]

Das Problem der Produzenten-Konsumenten-Koordination über einen beschränkten Puffer soll ein weiteres Mal angegangen werden. Diesmal soll nur eine eingeschränkte Form von Semaphoren zur Verfügung stehen, nämlich sogenannte *binäre* Semaphore. Diese sind gegenüber der allgemeinen Form eingeschränkt, indem sie nur *zwei Werte* annehmen können: 0 und 1.[2] Geben Sie eine Lösung für das Problem mit beschränktem Puffer aus Aufgabe 2, die mit binären Semaphoren auskommt.

[2] Läßt man negative Werte zu, müßte man präziser sagen, binäre Semaphore lassen zwei nicht-negative Werte zu, aber das spielt für die Aufgabe keine Rolle.