VO 182.711 / 182.737 8. April 2022

Prüfung Betriebssysteme

KNr. MNr. Zuname, Vorname

Studium: ☐ Informatik (Bakk.) ☐ Elektrotechnik (Master)

Ges.)(100)	1.)(30)	2.)(25)	3.)(45)	Zusatzblätter:

Bitte verwenden Sie nur dokumentenechtes Schreibmaterial!

1 Synchronisation mit Semaphoren (30)

In einer Lackierkammer werden Bahnwaggons mit Roboterunterstützung lackiert. Zwei Prozesse kontrollieren den Lackiervorgang, bei dem Waggons einzeln lackiert werden.

- Der Prozess *main_control* bringt Waggons in die Lackierkammer (*load_waggon()*) und transportiert sie nach dem Lackieren aus der Kammer (*remove_waggon()*).
- Der Prozess *paint_robot* führt die einzelnen Schritte des Heißlackierens durch einen Lackierroboter durch: der Roboter füllt zunächst seinen Farbtank an der Befüllungsanlage (*fill_paint()*), heizt dann den Farbwärmer auf (*warm_up()*), lackiert den Waggon (*paint()*) und kühlt am Ende den Farbwärmer ab (*cool_down()*).

Die Aktionen der Prozesse und die Synchronisation zwischen den Prozessen sehen Sie in den folgenden Programmstücken.

```
void main_control(void)
{
  while(test_work_to_do())
  {
    load_waggon();
    V(waggon_ready);
    /* waggon painting */
    P(waggon_painted);
    remove_waggon();
  }
}
```

```
void paint_robot(void)
{
  P(waggon_ready);

  fill_paint();
  warm_up()
  paint();
  cool_down();

  V(waggon_painted);
}
```

Um das Lackieren zu beschleunigen, wird der Roboterprozess durch drei parallele Roboterprozesse ersetzt.

Fügen Sie in den Code-Skeletten für die Roboterprozesse *Semaphoroperationen* ein, um die Roboterprozesse mit dem Prozess *main_control* und untereinander zu synchronisieren. Geben Sie geeignete Initialisierungen für alle Semaphore an. Ihre Synchronisationskonstrukte sollen folgende Punkte sicherstellen:

- Der Code von *main_control* bleibt unverändert.
- Die Roboter beginnen erst dann mit ihren Aktionen, wenn ein Waggon in der Lackierkammer ist. Der Abtransport des Waggons darf erst beginnen, wenn alle drei Roboter mit all ihren Arbeitsschritten fertig sind.
- An der Farbbefüllungsanlage kann zu jedem Zeitpunkt maximal ein Roboter aufgetankt werden. Die Befüllungsreihenfolge der Roboter soll nicht künstlich durch die Software eingeschränkt werden.
- Die Roboter beginnen gemeinsam mit dem Lackieren, wenn alle Roboter mit dem Aufheizen der Farbvorwärmer fertig sind. Das Lackieren durch die Roboter soll dann parallel erfolgen.

Initialisierungen:

```
void paint_robot1(void)
{

  fill_paint();

  warm_up();

  paint();

  cool_down();

}
```

```
void paint_robot2(void)
{

  fill_paint();

  warm_up();

  paint();

  cool_down();

}
```

```
void paint_robot3(void)
{

  fill_paint();

  warm_up();

  paint();

  cool_down();

}
```

2 Deadlock Avoidance – Banker's Algorithm (25)

In einem Computersystem gibt es drei Prozesse (P_1, P_2, P_3) und drei Arten von Ressourcen (R_1, R_2, R_3). Der *Resource Vector* R=(7, 8, 7) beschreibt die Anzahl der vorhandenen Ressourcen. Prozesse verwenden die Operationen *get*(r_1, r_2, r_3) bzw. *free*(r_1, r_2, r_3), um r_i Ressourcen der Ressourcenart R_i ($i = 1 \ldots 3$) anzufordern bzw. freizugeben.

Die folgende Abbildung zeigt für jeden der drei Prozesse die Folge von *get* und *free*-Operationen, die bei jeder Prozessabarbeitung durchgeführt werden.

P_1	P_2	P_3
get (2, 1, 1)	get (1, 1, 1)	get (0, 2, 2)
free (2, 1, 0)	get (1, 0, 0)	free (0, 1, 0)
get (1, 2, 2)	free (1, 1, 1)	get (1, 1, 1)
get (2, 1, 0)	get (0, 3, 0)	free (0, 0, 1)
free (2, 1, 1)	get (1, 0, 0)	get (0, 0, 1)
get (1, 0, 3)	free (1, 0, 0)	get (3, 1, 0)
free (1, 2, 2)	get (1, 0, 0)	free (2, 2, 2)
free (1, 0, 3)	free (2, 3, 0)	free (2, 1, 1)

Nehmen Sie an, dass die drei ersten Operationen jedes Prozesses abgearbeitet wurden (d.h., die Abarbeitung jedes Prozesses befindet sich an der strichlierten Linie). Als nächster Schritt soll die vierte Operation von P_1 (grau hinterlegte Operation) durchgeführt werden.

Verwenden Sie den Banker's Algorithmus, um festzustellen, ob die vierte Operation von P_1 durchgeführt werden soll. Geben Sie alle erforderlichen Vektoren und Matrizen an und führen Sie den Banker's Algorithmus schrittweise durch, d.h. geben Sie für jeden Schritt die Werte der Elemente aller relevanten Matrizen und Vektoren an.

3 Fragen zu Betriebssystemen (45)

Was versteht man unter einem *Process Image*? Erklären Sie, aus welchen Teilen ein Process Image besteht? (4)

Skizzieren Sie die Folge der Schritte, die bei der Abarbeitung eines *Synchronen I/O Requests* ablaufen. (5)

Zeichnen Sie das Zustandsdiagramm eines Prozesses, beginnend mit der Entstehung, bis zur Beendigung des Prozesses. Geben Sie die Namen der Zustände an und beschreiben Sie kurz jeden Zustand und Zustandsübergang. (5)

Nennen Sie Möglichkeiten, um in einem Paging-System Speicherschutz zu realisieren? (3)

Erklären Sie die Begriffe *absoluter Pfadname* und *relativer Pfadname* und geben Sie jeweils ein Beispiel an. (2)

Welcher Vorteil ergibt sich aus der Einführung von *Threads* für den Benutzer? Wodurch kommt es zu diesem Vorteil? Worauf muss der Benutzer bei der Programmierung von Threads achten? (3)

Bei der Realisierung von Dateisystemen gibt es verschiedene Möglichkeiten, um die zu einer Datei gehörenden Datenblöcke zu organisieren bzw. auffindbar zu machen (Block-Allokierung). Nennen Sie vier verschiedene Strategien zur Block-Allokierung von Dateien und beschreiben Sie diese mit ihren Vor- und Nachteilen. (5)

Nennen Sie vier Designprinzipien für die Konstruktion von sicheren Systemen. Geben Sie für jedes Prinzip ein Beispiel an. (4)

Beschreiben Sie die folgenden Strategien für das CPU Scheduling und vergleichen Sie deren Eigenschaften: *FCFS*, *Round Robin*, *SPN* und *SRT*. (4)

Beschreiben Sie Aufgabe und Funktion eines *Translation Lookaside Buffers*? Worauf hat man bei der Betriebssystemimplementierung bei einem Process Switch zu achten, wenn man einen Translation Lookaside Buffer verwendet? (3)

Mit welcher logischen Struktur des I/O Systems versucht man bei der Realisierung von I/O Funktionen sowohl ein einheitliches Programmierinterface, als auch eine möglichst gerätespezifische Ansteuerung zu erreichen? (4)

Beschreiben Sie das typische Layout einer Disk bzw. eines Filesystems. Erklären Sie kurz die einzelnen Komponenten. (3)