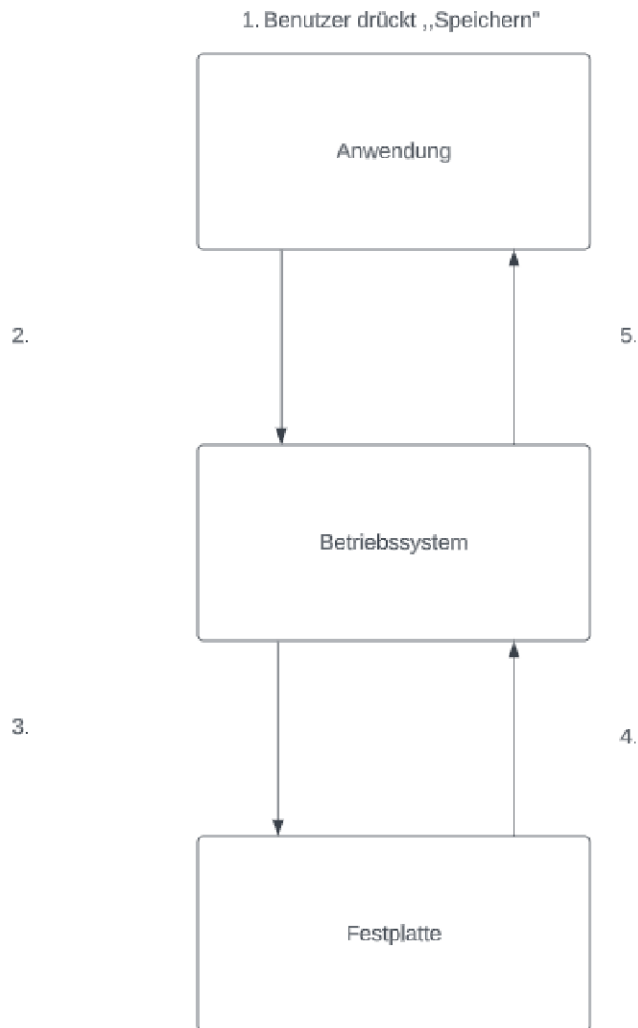


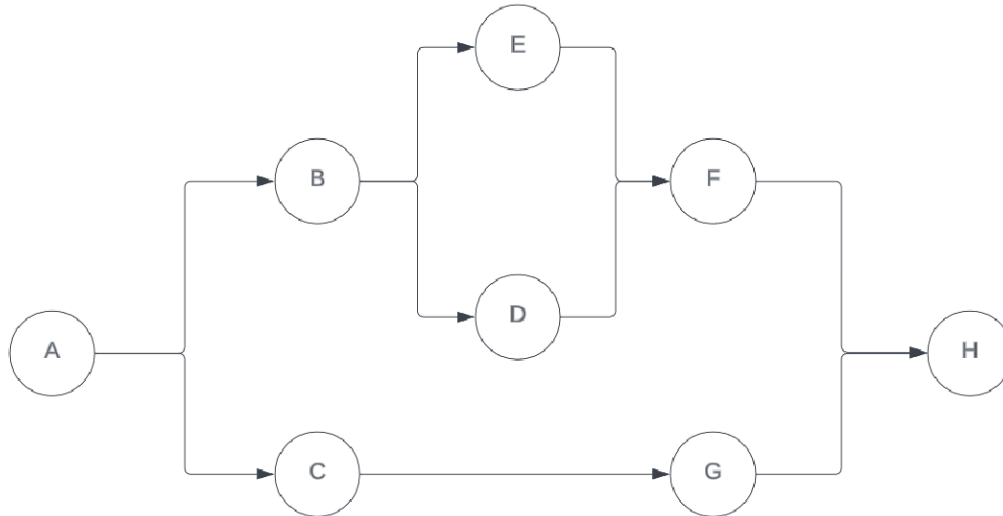
SysProg SoSe 2023 Klausur 1

Aufgabe 1 - Prozesse

(a) Beschriften Sie die Schritte in der Abbildung und beschreiben Sie sie jeweils unterhalb der Abbildung. (4p)



(b) Gegeben sei der folgende Abhängigkeitsgraph. Erstellen Sie Pseudocode für fork/join sowie parbegin/parend, der die Abhängigkeiten der Prozesse akkurat widerspiegelt. (6p)



Aufgabe 2 - Scheduling (12p)

- (a) Betrachten Sie folgende Tabelle ankommender Prozesse.
Nachdem Prozess A (Ankunftszeit 0; Länge 5 Einheiten) beendet wurde, wird anhand verschiedener Scheduling-Strategien ein nächster Prozess gewählt. Füllen Sie die Tabelle 2.2. (4p)

Prozess	Ankunftszeit	Länge
B	5	2
C	3	7
D	7	5
E	2	3

Tabelle 2.1: Prozessliste

Methode	LCFS-NP	SJN	FCFS	HRRN
Nächster Prozess				

Tabelle 2.2

(b) Handsimulationen (8p)

Simulieren Sie die drei Verfahren MLF ($t_i = 2^i$; $i = 0, 1, \dots$), LCFS-NP, und SJN per Hand. Ermitteln Sie für LCFS-NP und SJN auch für jeden Prozess die entsprechende Warte- und Antwortzeit.

MLF:

Name	Ankunftszeit	Länge
A	0	4
B	1	2
C	5	3
D	7	1

Tabelle 2.3

Zeit	0	1	2	3	4	5	6	7	8	9
CPU										
LVL 0										
LVL 1										
LVL 2										
LVL 3										

Tabelle 2.4

SJN:

Name	Ankunft	Länge
A	1	3
B	3	2
C	4	9
D	7	7
E	12	6

Tabelle 2.5

Zeit	1	2	3	4	5	6	7	8	9	10	11	12	13
CPU													
Queue													

14	15	16	17	18	19	20	21	22	23	24	25	26	27

Tabelle 2.6

Prozess	A	B	C	D	E
Wartezeit					
Antwortzeit					

Tabelle 2.7

LCFS-NP:

Name	Ankunft	Länge
A	1	3
B	3	2
C	4	9
D	7	7
E	12	6

Tabelle 2.8

Zeit	1	2	3	4	5	6	7	8	9	10	11	12	13
CPU													
Queue													

14	15	16	17	18	19	20	21	22	23	24	25	26	27

Tabelle 2.9

Prozess	A	B	C	D	E
Wartezeit					
Antwortzeit					

Tabelle 2.10

Aufgabe 3 - Koordination

- (a) Gegeben sei folgender Pseudocode. Die Label P1 und P2 werden jeweils von zwei Prozessen nebenläufig ausgeführt. Welche Werte kann x nach der Beendigung beider Prozesse annehmen? (Hinweis: Es handelt sich um **nicht** atomare Instruktionen) (3p)

```
int x = 3;  
  
P1:  
  x = x + 5;  
  
P2:  
  x++;
```

- (b) Im folgenden Pseudocode soll mithilfe von einem Mutex L und einem Signal C sichergestellt werden, dass bei nebenläufiger Ausführung höchstens zwei Prozesse gleichzeitig die Funktion getA() aufrufen können. Erklären Sie **kurz**, warum es hier zu einem Deadlock kommen kann und ändern Sie den Code, sodass dies nicht mehr passieren kann. (4p)

```
int count = 0;
mutex_t L;
signal_t C;

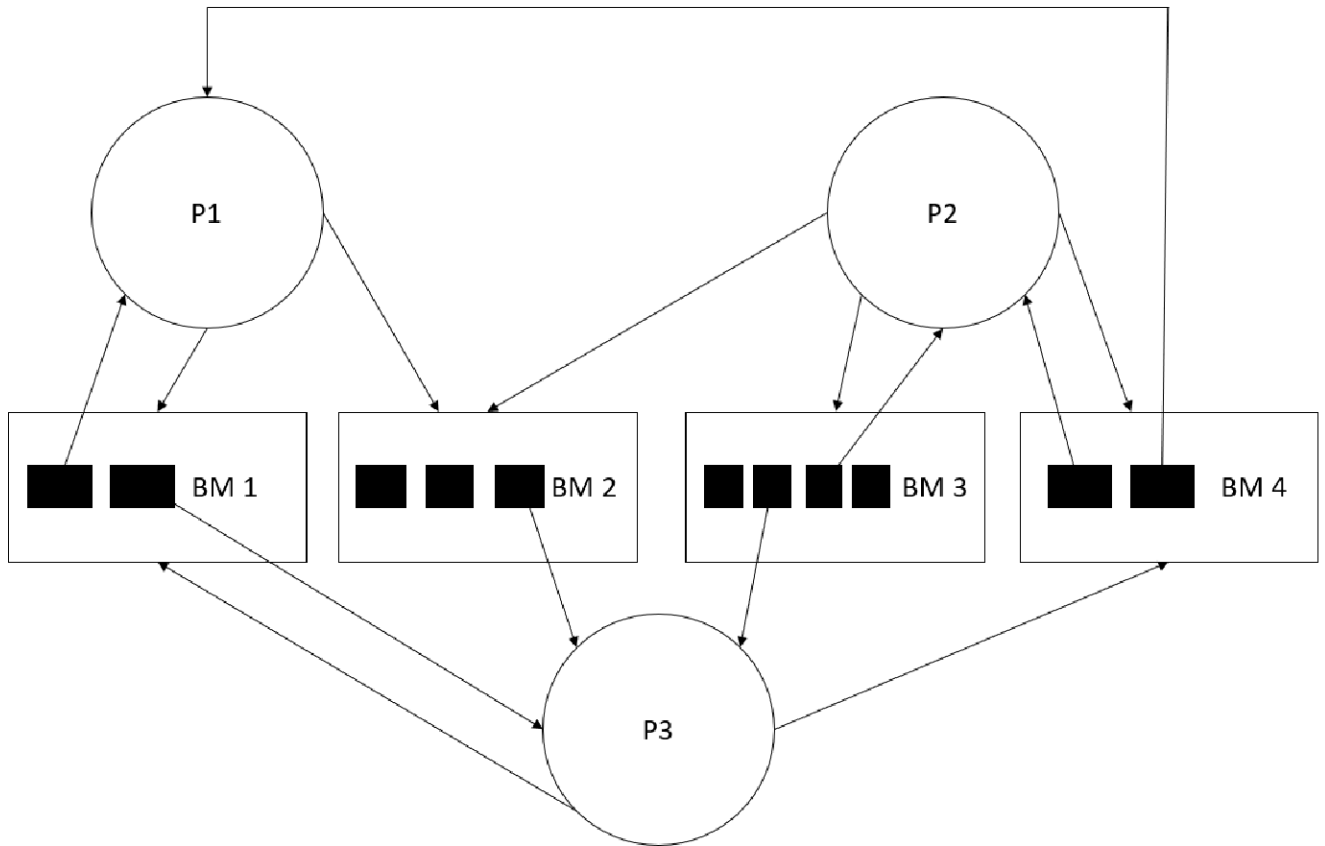
lock(L);
while (count >= 2) {
    wait(C);
}
count++;
unlock(L);

// Called by 2 threads at maximum (at the same time)
getA();

lock(L);
count--;
signal(C);
unlock(L);
```


4. Aufgabe - Betriebsmittelverwaltung

Gegeben sei der folgende Betriebsmittelgraph. Geben Sie die Matrizen der Belegung \mathbf{B} , der Restanforderungen \mathbf{R} , der Gesamtanforderungen \mathbf{G} sowie die Vektoren der freien Betriebsmittel \mathbf{f} und der grundsätzlich verfügbaren Betriebsmittel \mathbf{v} an. (4p)



5. Aufgabe - Speicherverwaltung

Die folgende Tabelle enthält eine SKT mit 2^4 Einträgen mit Tabellenbasisadresse 0x57. Sie befindet sich in einem Speicher mit 2^8 Einträgen. Übersetzen Sie die gegebenen virtuellen Adressen mithilfe der Seitentabelle in physische Adressen. (4p)

(Hinweis: Die gegebenen Werte für diese Aufgabe stimmen nicht mit den eigentlichen Werten aus der Klausur überein. Die Aufgabe an sich ist jedoch die Gleiche)

Adresse	Wert	Adresse	Wert
0x57	0x0C	0x5F	0x00
0x58	0x02	0x60	0x07
0x59	0x00	0x61	0x04
0x5A	0x08	0x62	0x09
0x5B	0x0F	0x63	0x09
0x5C	0x0B	0x64	0x00
0x5D	0x0B	0x65	0x01
0x5E	0x01	0x66	0x0D

Tabelle 5.1: Eine SKT

Virtuelle Adresse	Physische Adresse
0x27	
0xC6	
0xF1	
0x06	

Tabelle 5.2

6. Aufgabe - Sicherheit und BS-Architekturen

(a) Erklären Sie **stichpunktartig**, was ein Pufferüberlauf ist und wieso Pufferüberläufe ein Sicherheitsproblem darstellen können. (3p)

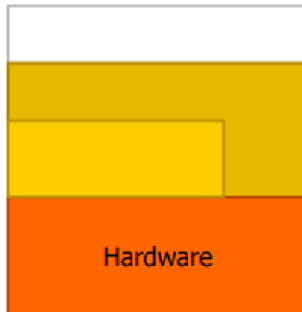
(b) Beschreiben Sie **kurz**, wie man als Programmierer Pufferüberläufe verhindern kann. (2p)

(c) Nennen Sie drei Vorteile von Mikrokern-Architekturen. (3p)

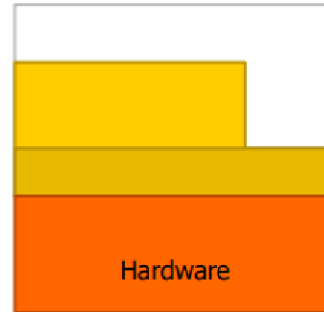
7. Aufgabe - Virtualisierung

(a) Beschriften Sie die folgenden Diagramme mit den Begriffen *Anwendung*, *Betriebssystem* und *Virtualisierungssoftware*. (3p)

Anwendungsvirtualisierung



Systemvirtualisierung



(b) Erläutern Sie in zwei Schritten, wie in einem virtualisierten Betriebssystem bei der Ausführung eines sensitiven Befehls mit *Trap & Emulate* vorgegangen wird. (2p)

1. Gast OS führt sensitive Instruktion aus
- 2.
- 3.