

### Arbeitsblatt „sequentielle Bearbeitung im Vergleich zu Multiprogramming“

- Gegeben: drei Aufträge A, B, C
- Jeder Auftrag führt hintereinander die folgenden Schritte aus:  
Dateneingabe 1 → warten, bis CPU frei → CPU-Benutzung 1  
→ Dateneingabe 2 → warten, bis CPU frei → CPU-Benutzung 2

- Zeiten für die einzelnen Schritte:

	Eingabe 1	CPU 1	Eingabe 2	CPU 2
Auftrag A	15	2	10	3
Auftrag B	10	6	5	2
Auftrag C	5	5	10	5

- Annahmen:
  - Umschalten der CPU zwischen Aufträgen kostet keine Zeit.
  - Prioritätsstufung:  $A > B > C$ 
    - Also bei sequentieller Bearbeitung:  
Erst A vollständig, dann B vollständig, dann C vollständig bearbeiten.
    - Also bei Multiprogramming:  
Von den Aufträgen, die die CPU gerade haben wollen, bekommt sie der höchstpriorität; niederpriorität Aufträge können von höherpriorität unterbrochen werden.
  - Bei Multiprogramming:
    - Auftrag braucht zur Eingabe die CPU nicht:  
Eingaben können unbeschränkt nebenläufig zu Aktionen anderer Aufträge erfolgen.
    - Wird ein Auftrag unterbrochen, so wird seine bisherige CPU-Benutzung angerechnet.
- Aufgaben: für sequentielle Bearbeitung und Multiprogramming jeweils
  - Zeitdiagramm zeichnen:  
Wann bedient die CPU welchen Auftrag und wann ist sie „idle“ (=unbeschäftigt)?
  - Berechnen:
    - Aufenthaltsdauern der Aufträge A, B und C  
(= Zeitdauer vom Zeitpunkt 0 bis zu ihrer jeweiligen Fertigstellung)
    - mittlere Aufenthaltsdauer  
(= arithmetisches Mittel der drei Aufenthaltsdauern)
    - Auslastung der CPU  
(= prozentualer Anteil der Zeit der Auftragsbearbeitungen an der Gesamtzeit)