

# Vorlesung Betriebssysteme 1

## Aufgabenblatt 4 zu Unit OS5

(Besprechung im Übungstermin gemäß Ankündigung auf der Vorlesungs-Website)

Die Aufgaben auf diesem und den folgenden Übungsblättern sollen Ihnen helfen, Ihr Selbststudium zu strukturieren und Ihnen ein Gefühl dafür geben, welchen Wissensstand Sie bisher in der Veranstaltung erlangt haben sollten. Um die gemeinsame Zeit mit den Tutoren optimal nutzen zu können, sollten Sie versuchen, bis zum jeweiligen Tutorengespräch möglichst viele der Aufgaben zu bearbeiten.

Die Bearbeitung sollte in einer Art und Weise erfolgen, dass a) Sie und Ihre Gruppe gut mit den Ergebnissen arbeiten können und b) sich im Gespräch mit den Tutoren über den Stoff der Aufgaben austauschen können. Es bestehen keine weiteren formalen Anforderungen an die Aufgabebearbeitung unsererseits.

1. Beschreiben Sie die Rolle der *Memory Management Unit* (MMU) im Betriebssystem. Was sind dabei *Virtuelle Speicheradressen* und *Physische Speicheradressen*?
2. Eine frühe Idee zur Verwaltung und Isolation der Speicherbereiche mehrerer Prozesse war *Segmentierung*. Beschreiben Sie die Funktionsweise von Segmentierung, und gehen Sie dabei auch auf die Probleme ein, die mit der Segmentierung des Speichers einher gehen.
3. Moderne Betriebssysteme benutzen *Paging* zur Verwaltung der Speicherbereiche von Prozessen. Beschreiben Sie die Funktionsweise von Paging, und gehen Sie dabei darauf ein, wie Paging die Probleme von Segmentierung behebt. Betrachten Sie auch die Nachteile, die dadurch entstehen. Beschreiben Sie außerdem die Funktionsweise des *Translation Lookaside Buffers* (TLB), und seine Rolle bei der Begrenzung des durch Paging entstehenden Overheads beim Speicherzugriff.
4. Beschreiben Sie den Aufbau einer typischen *Seitentabelle* (*Page Table*). Nennen Sie typische *Flaggen*, die zusammen mit Einträgen in der Seitentabelle gespeichert werden, und erläutern Sie deren Funktion.

5. Was ist das *Working Set* eines Prozesses?
6. Betrachten Sie ein Rechnersystem welches 26-Bit breite Adressen benutzt, und in einem einstufigen Verfahren Speicherseiten mit einer Größe von 256 Byte verwaltet. Beantworten Sie zu diesem Rechnersystem die folgenden Fragen:
  - (a) Wieviele Einträge hat eine Seitentable in diesem System?
  - (b) Was ist die Größe eines virtuellen Adressraums auf diesem System?
  - (c) Nehmen Sie an, dass zu jedem Seitentabelleneintrag 2 Statusbits gespeichert werden. Was ist die Größe eines Page Table Eintrags in Bits?
  - (d) Nehmen Sie an, dass das System 26 Bit Worte verwendet, und dass jeder Seitentabelleneintrag in der Seitentabelle auf Wortgröße aufgerundet wird. Wie groß ist dann eine gesamte Seitentabelle?
  - (e) Nehmen Sie weiterhin an, dass das System 24MiB ( $24 * 1024 * 1024$  Bytes) Hauptspeicher zur Verfügung hat, wovon das Betriebssystem 5MiB für Speichertabellen reserviert hat. Wieviele Prozesse kann dieses Betriebssystem maximal unterstützen?
  - (f) Nehmen Sie weiterhin an, dass das Betriebssystem zusätzlich zu den für Speichertabellen reservierten 5MiB weitere 6MiB Hauptspeicher für Kernel Datenstrukturen reserviert hat. Was ist damit die maximale Größe des *Working Sets* eines Prozesses auf diesem System?
7. Was ist ein *Seitenzugriffsfehler (Page Fault)*? Unterscheiden Sie dabei zwischen *Soft Page Faults* und *Hard Page Faults*.
8. Beschreiben Sie die Rollen der *Modified*, *Standby*, *Free*, *Zero* und *Bad Page List* sowie der *Page Frame Number Database* in der Speicherverwaltung des Windows Betriebssystems, und welche Eigenschaften Speicherseiten in den jeweiligen Listen haben.
9. Grenzen Sie die Begriffe *Swapping* und *Page Replacement* voneinander ab.
10. Beschreiben Sie den Unterschied zwischen *Reserved Memory* und *Committed Memory* in Windows.
11. Beschreiben Sie, wie Betriebssysteme die Page Tables in mehreren Prozessen verwenden können, um *Shared Memory* zu implementieren.
12. Was ist *Copy-on-Write* Speicher, und wofür wird er benutzt?
13. Was ist eine *Guard Page*?

14. Beschreiben Sie die *Seitenersetzungs-Algorithmen* (*Page Replacement Algorithmen*) *First-In-First-Out* (FIFO), *Second Chance* und *Least-Recently-Used* (LRU). Betrachten Sie dazu ein fiktives Rechnersystem mit 3 physischen Speicherseiten und dem folgenden Zugriffsmuster auf die virtuellen Speicherseiten:

2, 4, 0, 3, 2, 4, 1, 2, 4, 0, 3, 1

Stellen Sie die Abläufe tabellarisch dar. Wie viele Seitenzugriffsfehler erzeugen die Algorithmen? Wie viele Speicherzugriffsfehler würden sie erzeugen, wenn das Rechnersystem 4 physische Speicherseiten anstatt 3 Speicherseiten zur Verfügung hätte?