

## Aufgabe 1

### Begriffe

1. Beschreiben Sie in kurzen Sätzen die Idee hinter den Protection Rings. Wieviele Ringe werden von gängigen Betriebssystemen zu welchem Zweck verwendet?
2. Gliedern Sie den Begriff Server in das Konzept der Server-Client Architektur ein.
3. Beschreiben Sie in wenigen Sätzen die vier in der Vorlesung vorgestellten Implementierungen von Systemaufrufen bei monolithischem Kernel und Microkernel in Hinsicht auf den Ablauf der Wechsel zwischen User- und Kernelspace.

### Textaufgabe

Was ist das Thema der Tanenbaum–Torvalds-Debatte? Was ist Ihr Standpunkt zu dem Thema? Begründen sie Ihren Standpunkt.

## Aufgabe 2

In dieser Aufgabe sollen sie nicht die `stdio.h` Bibliothek verwenden. Führen Sie Fehlerbehandlung durch und erläutern Sie im Quellcode (als Kommentar) warum Sie sich für genau diese Fehlerbehandlung entschieden haben. Ihr Programm muss mit folgenden Compilerflags ohne Warnungen und / oder Fehler compilieren:

```
# gcc -std=c11 -o <program.out> -Wall -Wextra -pedantic <program.c>
```

1. Machen Sie sich mit den Linux Low-Level-IO system calls `open()`, `read()` und `write()` vertraut (Stichwort man pages). Suchen Sie weiterhin system calls zum Erstellen von Ordnern und Löschen von Dateien heraus.
2. Schreiben Sie eine Funktion mit der Signatur `int copy(char *sourcename, char *targetname)`, die den Inhalt einer Datei dann und nur dann in eine Zieldatei kopiert, wenn die Zieldatei noch nicht existiert.
3. Implementieren Sie unter Zuhilfenahme ihrer `copy` Funktion einen einfachen kommandozeilenbasierten Papierkorb. Das Programm soll bei Ausführung dann und nur dann im aktuellen Verzeichnis den Unterordner `.ti3_trashcan` erstellen, wenn dieser Ordner noch nicht existiert. Folgende Funktionalitäten sollen implementiert werden:
  - (a) DELETE  
Der Aufruf `./trashcan -d filename` verschiebt die Datei 'filename' in den versteckten Ordner `.ti3_trashcan` (unter Linux ist ein Ordner versteckt, wenn er mit einem Punkt beginnt).
  - (b) RECOVER  
Der Aufruf `./trashcan -r filename` verschiebt die Datei 'filename' aus dem versteckten Ordner `.ti3_trashcan` in das aktuelle Verzeichnis.
  - (c) FINALLY DELETE  
Der Aufruf `./trashcan -f filename` löscht die Datei 'filename' aus dem versteckten Ordner `.ti3_trashcan`.