

*Bitte beachten Sie die allgemeinen Hinweise auf Übungszettel 1*

## Aufgabe 1: Memory

1. Erklären Sie die folgenden Begriffe mit je einem Satz:
  - (a) Thrashing,
  - (b) TLB,
  - (c) MMU,
  - (d) physikalische Adresse,
  - (e) logische (auch virtuelle) Adresse und
  - (f) Adressraum.
2. Erläutern Sie die Problematik von interner und externer Fragmentierung an Hand des Beispiels aus der Vorlesung. Wie löst Paging diese?
3. Ein Programm, das direkt nach dem Starten die Adresse der Main-Funktion ausgibt und dann in eine Endlosschleife verfällt, wird dreimal parallel gestartet. Was wird ausgegeben und warum?

## Aufgabe 2: Speicherverwaltung

Entwickeln Sie eine eigene Speicherverwaltung. Funktionalitäten zur Speicherverwaltung aus Bibliotheken sollen (außerhalb der `memory_init()` Funktion) nicht genutzt werden (kein `malloc()` `calloc()` oder ähnliches). Geben Sie die Ausgaben auf `stdout` aus, die Fehler auf `stderr`.

1. Schreiben Sie folgende Funktionen
  - (a) `void memory_init()`  
Initialisiert den zur Verfügung stehenden Speicherbereich und etwaige Verwaltungsdaten.
  - (b) `void* memory_allocate(size_t byte_count)`  
Gibt einen Pointer auf zusammenhängenden Speicherbereich der Größe `byte_count` zurück
  - (c) `void memory_free(void* pointer)`  
Gibt einen von `memory_allocate` reservierten Speicherbereich wieder frei
  - (d) `void memory_print()`  
Visualisiert den aktuellen Zustand des Speichers
2. Simulieren Sie den Hauptspeicher durch ein char-Array `memory` der Größe `MEM_SIZE`.
3. Überlegen Sie sich eine geeignete Struktur für Ihre Daten (Hinweis: verkettete Liste<sup>1</sup>, Structs).
4. Überlegen Sie sich Testfälle für Ihre Funktionen und implementieren Sie diese. Welche Sonderfälle sind zu berücksichtigen?

---

<sup>1</sup>Eine einfach verkettete Liste (linked list) ist eine dynamische Datenstruktur und besteht aus einer Menge von Knoten, die untereinander verkettet sind. Jeder Knoten besteht aus dem zu speichernden Objekt und einem Zeiger auf das nächste Element. Das letzte Element der Liste zeigt auf NULL und mittels eines Zeigers `head` wird auf den ersten Knoten in der Liste gezeigt.